

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Informační systém pro firmu Star Communications

Information system for Star Communications company

2010

Jan Uhlář

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2010

.....

Rád bych na tomto místě poděkoval všem, kteří mi s prací pomohli, protože bez nich by tato práce nevznikla. Jmenovitě především panu Ing. Lumíru Návratovi za vedení mé diplomové práce. Dále pak své přítelkyni a rodině, kteří mi byli oporou.

Abstrakt

Firma Star Communications a.s. se stala oficiálním partnerem společnosti Vodafone Czech Republic pro distribuci předplacených služeb. Prozatím měla informační systém, který umožňoval pouze statické procházení stránek. S přibývajícím počtem partnerů vzrůstá i nutnost jejich spravování a organizování. Dále firma potřebuje umožnit distribuci produktů na internetu a on-line nakupování. Výše zmíněné potřeby vedly k nutnosti vytvořit informační systém, který bude efektivní, umožní on-line registraci partnerů, spravování uživatelů a provozoven, nakupování zboží přes e-shop a CRM systém pro komplexní řízení.

Klíčová slova: Ruby, Rails, MySQL, internetový portál, e-shop, CRM systém, diplomová práce

Abstract

Company Star Communications corp became an official partner of Vodafone Czech Republic for the distribution of prepaid services. For now had an information system that allowed only static browsing. With increasing number of partners has also increased the need for the administration and organization. The company also needs to allow the distribution of products on the Internet and on-line shopping. The above mentioned needs have led to establish an information system that is efficient, allowing on-line registration partners, manage users and offices, purchasing goods via the e-shop and CRM system for comprehensive management.

Keywords: Ruby, Rails, MySQL, internet portal, e-shop, CRM system, master thesis

Seznam použitých zkratek a symbolů

AJAX	– Asynchronous JavaScript and XML
CMS	– Content Management System
CRM	– Customer Relationship Management
CSV	– Comma-separated Values
DFD	– Data Flow Diagram
HTML	– Hyper Text Markup Language
MVC	– Model, View, Controller
POS	– Point Of Sale
SQL	– Structured Query Language
SVN	– Subversion
UML	– Unified Modeling Language
URL	– Unique Resource Locator
WYSIWYG	– What You See Is What You Get
XML	– Extensible Markup Language

Obsah

1 Úvod	6
2 Specifikace požadavků	7
2.1 Z pohledu webové části	7
2.2 Z pohledu CRM části	9
3 Stakeholders, aktéři a cíle	11
3.1 Neregistrovaný uživatel	13
3.2 Partner - administrátor	13
3.3 Partner - uživatel	13
3.4 CRM - administrátor	14
3.5 CRM - obchodník	16
3.6 CRM - marketing	16
4 Analýza a návrh	21
4.1 Webová část	21
4.2 CRM část	25
5 Implementace	32
5.1 Ruby	32
5.2 Rails	33
6 Nasazení a testování	41
6.1 Manuální testování	41
6.2 Funkcionální testování	42
6.3 Výkonnostní testování	44
6.4 Závěrečná zpráva	45
7 Závěr	47
8 Reference	48
Přílohy	48
A Obsah DVD	49
B Dokument vize	50
B.1 Předmluva	50
B.2 Nastavení	51
B.3 Stakeholders a uživatelský popis	52
B.4 Přehled o produktu	55
B.5 Funkce produktu	55
B.6 Omezení	58
B.7 Rozsah kvality	59

B.8	Priority	59
B.9	Dokumentace	59

Seznam tabulek

1	Úryvek z glossary	8
2	Stakeholders	12
3	Aktéři	12
4	UC – Správa partnerů	18
5	Categories	36
6	Items	36
7	Categories_items	36
8	Obsah DVD	49
9	Přehled stakeholderů	53
10	Přehled uživatelů	53
11	Uživatelské potřeby	56
12	Souhrn schopností	57

Seznam obrázků

1	Diagram užití: Správa partnerů	19
2	Diagram aktivit: Správa partnerů	20
3	Kontextový diagram	22
4	DFD – 0. úroveň	22
5	Třídní diagram: Registrace partnera	24
6	Třídní diagram: On-line nakupování	25
7	Doplňný třídní diagram: Správa partnerů	26
8	Stavový diagram: Objednávka	27
9	Sekvenční diagram: Vytvoření produktu	28
10	Třídní diagram: Hromadné emaily	29
11	DFD_1: Zápisy z jednání	30
12	Třídní diagram: Zápisy z jednání	31
13	MVC a Helper – předávání dat	34

Seznam výpisů zdrojového kódu

1	Ruby – vše je objekt	33
2	Program v jazyce Ruby	33
3	Rails – Active Record	35
4	Příklad použití validace	36
5	Příklad ActiveRecord	37
6	Příklad na ActionController	38
7	Ukázka routování	38
8	Ukázka ActiveSupport	39
9	Ukázka ActionView	39

1 Úvod

Firma, která má mnoho klientů a nabízí jim své produkty, potřebuje určitým způsobem zajistit jejich správu a distribuci. Evidence záznamů v podobě složek a papírů není moc efektivní a proto se nabízí možnost použití počítačového řešení ve formě internetové aplikace. Internet je v dnešní době nejčastějším nástrojem pro vyhledání informací a také pro vlastní zviditelnění.

Internetové aplikace nám umožňují nalézt odpovědi na různé otázky, pomáhají nám při výběru zboží, při hledání informací v diskusních fórech, usnadňují nám život v podobě e-shopů, protože nemusíme chodit do kamenného obchodu a obecně se staly významnou součástí našich životů.

Proto se firma Star Communications a.s. rozhodla pro tvorbu internetového portálu, který bude splňovat jejich požadavky a usnadní jim tak práci. V následujícím textu bude popsáno jak postupovat při vytváření takového portálu.

V další kapitole vás seznámím se specifikací požadavků. Kapitola č. 3 se zabývá všemi aktéry, stakeholdery a jejich cíli. V kapitole č. 4 se zaměřím na analýzu a návrh. Kapitola č. 5 nás seznámí s programovacím jazykem Ruby a frameworkem Rails. A kapitola č. 6 je zaměřena na testování a nasazení aplikace do provozu.

2 Specifikace požadavků

Je nereálné zachytit všechny požadavky před začátkem vývoje softwarového díla.

V průběhu práce na systému se požadavky mění a vznikají nové. Identifikování a porozumění požadavkům je opakující se proces. V této kapitole se pokusím zachytit všechny požadavky, které byly v průběhu práce na systém kladeny.

Ve fázi specifikace požadavků také vznikají různé dokumenty na podporu systému. Jedná se zejména o glosář (glossary), který slouží jako slovník mezi zadavatelem a zhotovitelem. V něm se definují běžně používané výrazy, kterým by koncový uživatel nemusel zcela rozumět. Slouží jako prevence proti nedorozumění. V tabulce 1 je zobrazena část slovníku.

Dále vytváříme tzv. dokument vize, který popisuje současný problém, celkovou vizi budoucího systému, definuje stakeholdery, uživatele a jejich požadavky a priority, dává nám přehled o budoucím systému, zobrazuje jeho hlavní vlastnosti, rozsahy a omezení. Odpovídá na otázky „co“ a „proč“, týkající se systému. Slouží také jako podklad pro use case modelování. Vzhledem k obsáhlosti jsem umístil dokument vize do přílohy B.

Abych mohl lépe popsat funkční požadavky, zaměřím se na popis tzv. příběhů z pohledu všech různých uživatelů systému. V těchto příbězích bude popsáno, co který uživatel od aplikace očekává. Sjednocením těchto popisů pak vznikne úplná funkční specifikace.

2.1 Z pohledu webové části

Z tohoto pohledu mohou přijít do styku se systémem tři různé skupiny uživatelů.

1. Neregistrovaný uživatel

Je potřeba, aby se na stránkách dalo s přehledem orientovat. Dále aplikace musí nabídnout registraci, při které se vyplní všechny potřebné údaje, aby se uživatel mohl stát partnerem firmy a měl možnost nakupovat POS materiály a jiné produkty.

Při registraci je potřeba zadat všechny informace o firmě a mít možnost hned v úvodu vytvořit všechny provozovny, které firma spravuje. Bylo by vhodné umožnit přidání informací o majiteli firmy.

Dále systém musí umožnit vytvořit uživatelský přístup (login a heslo), a nakonec se musí zobrazit shrnutí registrace se všemi vyplněnými údaji, které uživatel zadal. Toto shrnutí bude sloužit pro kontrolu, zda někdo neudělal chybu ve vyplňování.

V případě výskytu chyby by měla existovat možnost vrátit se zpět, odstranit chybu, ale všechny ostatní údaje by měly zůstat předvyplněné, aby je uživatel nemusel zadávat znovu. Po dokončení registrace musí přijít potvrzovací email s vyplněnými údaji a přístupovým heslem, aby jej partner nezapomněl.

Systém by měl na úvodní stránce umožnit snadnou orientaci při hledání nabízených produktů a firemních informací.

Termín	Význam	Alias
abstraktní	Opak konkrétního	
aktér	Někdo nebo něco vně systému, co je v přímé interakci se systémem	
asociace	Vztah, který modeluje spojení mezi instancemi	
atribut	Atribut definovaný třídou reprezentuje název proměnné a má určitý typ	
developer	Člověk, který je zodpovědný za implementaci žádané funkcionality	návrhář
diagram	Grafické znázornění části systému	
element	Nedělitelná složka modelu	
FURPS	Akronym, který znázorňuje kategorie pro zjišťování kvality produktu: Functionality, Usability, Reliability, Performance, Supportability.	
heslo	Sekvence minimálně šesti znaků, která slouží pro přihlášení uživatele spolu s položkou login.	
instance	Individuální entita, která má určitý stav a mohou být na ní aplikovány operace	
iterace	Sekvence aktivit	
komponenta	Část systému, která je nezávislá na okolním dění a vykonává danou funkci. Fyzická reprezentace zdrojového kódu	
košík	Elektronická úschovna položek pro následné objednání. Obsah košíku je uložen v sessions.	nákupní košík
login	Unikátní sekvence znaků, která slouží pro rozlišení uživatelů. Slouží také jako přihlašovací údaj.	
návrh	Část softwarového procesu, která se zaměřuje na to, jak bude systém implementován	
objekt	Entita s jednoznačně danými hranicemi a identifikací, která má své chování a vlastnosti	
repository	Úložiště pro modely, rozhraní a implementaci	
session	Speciální paměť pro uložení informací. Např. login, heslo, košík	
stakeholder	Osoba, která je podstatně ovlivněna výsledkem systému	
třída	Množnina všech objektů, které mají stejné atributy, vlastnosti	
uživatel	Osoba nebo entita, která přímo používá systém nebo výstup systému.	
validní	Odpovídající určitým normám. Např. validní IČO je osm číslic.	

Tabulka 1: Úryvek z glossary

2. Registrovaný partner - administrátor

Jako vedoucí firmy, která se stala partnerem, potřebuje mít více práv pro práci se systémem, než ostatní zaměstnanci firmy. Po přihlášení by měl systém rozpoznat, že je uživatel vedoucí a měl by mu zpřístupnit možnost vytvoření nové objednávky, kde by si zvolil počet kusů, na kterou provozovnu chce objednávku doručit a po potvrzení odeslat informační email.

Dále je potřeba mít přehled o historii všech objednávek. Pokud by partner chtěl přiložit různé soubory, týkající se jeho firmy, měla by tato možnost být jednoduše přístupná. V budoucnosti by možná chtěl i soubory mazat nebo upravovat.

Systém také musí nabídnout možnost měnit osobní údaje a informace o firmě a provozovnách. Jednoduchým způsobem musí umožnit správu ostatních kontaktních osob, nastavení jejich práv a jejich přiřazení k libovolné provozně.

3. Registrovaný partner - uživatel

Až na poslední odstavec by měl mít v systému stejná práva jako partner - administrátor.

2.2 Z pohledu CRM části

1. CRM - administrátor

Pro dokonalé vedení firmy je potřeba mít v aplikaci dostupný CRM systém, se kterým bude možné intuitivně pracovat. Jedním ze základních prvků tohoto systému musí být správa objednávek. Aplikace musí mít komplexní formulář pro vyhledání objednávek podle různých kritérií. Po potvrzení vyhledávacích kritérií systém zobrazí seznam nalezených objednávek po určitém počtu na stránce s údaji o partnerovi, datu vytvoření a stavu objednávky. V tuto chvíli je k dispozici volba okamžitého označení objednávky za vyřízenou nebo se dá prokliknout na detail, kde je možné označit objednávku za zrušenou, nebo vytvořenou s možností poslat odpovídající potvrzovací email.

Dalším stavebním kamenem je správa partnerů. Musí zde být také komplexní formulář pro vyhledání a třídění partnerů podle různých kritérií. Výpis by měl být opět stránkovaný s údaji o datu registrace, datu poslední aktivity a jeho obchodním zástupci. Zde musí být možnost změny obchodního zástupce, možnost okamžitého přidání zápisu z jednání, nebo zobrazení detailu partnera.

Když si uživatel zobrazí detail partnera, může editovat údaje o firmě, kontaktních osobách a provozovnách. Dále systém musí umožnit vytvoření nové objednávky, včetně možnosti upravit ceny pro konkrétního partnera a zobrazení historie objednávek. V menu nesmí chybět volba přidání zápisu z jednání a zobrazení všech událostí a zápisů z jednání s možností filtrování pomocí formuláře, ve kterém je mimo jiné i volba hledání podle data, která je velice důležitá.

Typy událostí mohou být přihlášení do systému, nahrání souboru, vytvoření objednávky a nebo poslání hromadného emailu. Zápisy z jednání je možné filtrovat

podle toho, zda mají přílohu, nebo podle typů (schůzka, email, telefonát, soutěž), které je možné také přidávat.

Aby mohl uživatel průběžně reagovat na změny u produktů (ceny, názvy, atd..), musí mít v CRM sekci, která by mu to umožnila. V této sekci také musí být správa kategorií, kde by mohl měnit název, jejich pozici a nastavení, zda je kategorie aktivní.

Protože nebude se systémem pracovat pouze jeden uživatel, musí zde být správa CRM uživatelů, která umožní nastavení loginu, jména, příjmení, emailu, hesla, oprávnění a zda je aktivní. Typy oprávnění jsou obchodní zástupce, marketing a administrátor.

Další důležitou vlastností tohoto systému je správa reportů. V této sekci musí být komplexní formulář pro vyhledání a třídění různých typů reportů. Typy reportů jsou Zápisy z jednání, Přehled objednávek, Přehled partnerů vč. kontaktů, Přehled partnerů - pobočky, Události, Přehled kontaktních osob. Jednotlivé reporty musí mít možnost exportování do formátu CSV nebo tisku.

Vzhledem k plánovanému množství partnerů je velice důležité, aby uživatel mohl vytvářet hromadné emaily. Pomocí wysiwyg editoru zadá text. Poté se zobrazí hrubý filtr příjemců podle krajů a následuje ruční filtr příjemců podle kontaktních osob, kde je možné nastavit oslovení 5. pádem. Další krok vede k rekapitulaci a následnému odeslání.

Někdy bude určitě potřeba vytvářet doplňující stránky, které by měly být snadno editovatelné pomocí wysiwyg editoru. U každé stránky musí být možnost zadat jméno, URL, nastavení, zda je aktivní a vlastní obsah.

Dále systém musí umožnit editaci emailových adres pro posílání nových objednávek, znění informačního emailu o vytvoření, potvrzení a zrušení objednávek a obecné oslovení (pro případ, že u kontaktní osoby není vyplněno oslovení 5. pádem).

Správa webových stránek umožní editování všech stránek pomocí wysiwyg editoru a vytváření (editaci) článků do sekcí média a tiskové zprávy, které se hojně uplatní.

Všechny tabulky, které se objeví na výstupu musí mít možnost exportu do CSV formátu a tisku.

2. CRM - obchodní zástupce

Obchodní zástupce má přístup pouze do sekcí Správa objednávek, Správa partnerů, Správa reportů a Statistiky.

3. CRM - marketing

Marketing má přístup do všech sekcí v CRM kromě správy webových stránek.

3 Stakeholders, aktéři a cíle

Na základě příběhů můžeme identifikovat skupiny stakeholderů a aktérů. Nejprve si určíme jaký je mezi nimi rozdíl.

Stakeholder je osoba, která je zainteresovaná do naší aplikace a poskytuje všechny důležité informace o systému. Může to být zákazník, uživatel, ale i ředitel, který se systémem pracovat vůbec nebude. Stakeholder nemusí do systému nikterak zasahovat. V tabulce 2 můžete vidět všechny stakeholders, kteří v systému existují.

Aktér reprezentuje roli, kterou může mít člověk, hardware nebo jiný software ve vztahu k systému. Kompletní seznam aktérů popisuje všechny možnosti, kterými lze komunikovat se systémem. Aktéři mají cíle podporované popsaným systémem, podílí se na práci se systémem a pomáhají nám určit hranice systému. V tabulce 3 můžete vidět všechny aktéry.

Jméno	Reprezentuje	Role
CRM - administrátor	Skupina lidí, kteří jsou určeni firmou Star Communications ke správě celého CRM systému.	Zodpovědný za aktuálnost informací v systému (ceny produktů, webové stránky, tiskové zprávy).
CRM - marketing	Zaměstnanci firmy Star Communications se všemi právy kromě editace webových stránek.	Spravuje partnery, objednávky, produkty, hromadné emaily, zápisy z jednání...
CRM - obchodník	Zaměstnanci Star Communications s omezenými právy.	Má přístup pouze k partnerům, objednávkám, reportům a statistikám.
Partner - administrátor	Registrovaná kontaktní osoba nebo vedoucí partnerské firmy.	Má přístup k celému webovému rozhraní. Může vytvářet další kontaktní osoby, provozny...
Partner - uživatel	Kontaktní osoba s omezenými právy. Má login a heslo.	Může se přihlašovat a vytvářet objednávky.
Nepřihlášený uživatel	Běžný člověk, který není přihlášený.	Má roli pozorovatele celého vnějšího systému. Může se registrovat.
Manažer	Vedení firmy	Řídí firmu jako celek. Určuje ceny, přijímá nové lidi.

Tabulka 2: Stakeholders

Jméno	Popis	Stakeholder
CRM - administrátor	Spravuje celý CRM systém.	CRM - administrátor.
CRM - marketing	Spravuje partnery, objednávky, produkty, atd.	CRM - marketing.
CRM - obchodník	Spravuje partnery, objednávky, reporty a statistiky.	CRM - obchodník.
Partner - administrátor	Má přístup k celé webové části.	Partner - administrátor.
Partner - uživatel	Má přístup pouze k objednávkám a souborům.	Partner - uživatel.
Nepřihlášený uživatel	Libovolný návštěvník portálu.	Nepřihlášený uživatel.
E-mail server	Stará se o posílání emailů	

Tabulka 3: Aktéři

Po shrnutí stakeholderů a aktérů se podívejme na jednotlivé uživatelské cíle, což jsou v podstatě jednotlivé funkce systému. Jednotlivým cílům bude odpovídat vždy minimálně jeden případ užití.

Use case neboli případ užití je postupný sled událostí, které se musí vykonat k provedení určité akce. Případ užití je inicializován určitým aktérem. Funkčnost celého systému je definována množinou všech případů užití, kde každý z nich reprezentuje specifickou posloupnost akcí.

Jejich grafické znázornění je snadno pochopitelné pro všechny uživatele. Případy užití pomáhají ověřit, zda máme zaznamenány veškeré požadavky a zda jim stakeholdeři rozumí.

3.1 Neregistrovaný uživatel

1. procházení portálu
2. komplexní registrace

3.2 Partner - administrátor

1. vytvoření nové objednávky
2. procházení historie objednávek
3. přiložení různých souborů, týkajících se firmy
4. změna svých osobních údajů
5. změna údajů o své firmě a provozovnách
6. vytváření a editování kontaktních osob
7. nastavování práv všem svým osobám
8. přiřazení kontaktní osoby k libovolné provozovně

3.3 Partner - uživatel

1. vytvoření nové objednávky
2. procházení historie objednávek
3. přiložení různých souborů, týkajících se firmy
4. změna svých osobních údajů

3.4 CRM - administrátor

1. Správa objednávek
 - (a) komplexní formulář pro vyhledání objednávek podle různých kritérií
 - (b) seznam vyhledaných objednávek po určitém počtu na stránce s údaji o partnerovi, datu vytvoření a stavu objednávky
 - (c) možnost okamžitého označení objednávky za vyřízenou nebo prokliku na detail
 - (d) možnost označení objednávky za zrušenou
 - (e) možnost poslat potvrzovací email o vyřízení nebo zrušení objednávky
2. Výpis partnerů
 - (a) komplexní formulář pro vyhledání a třídění partnerů podle různých kritérií
 - (b) seznam vyhledaných partnerů po určitém počtu na stránce s údaji o datu registraci, datu poslední aktivity a jeho obchodním zástupci
 - (c) možnost změnit obchodního zástupce
 - (d) možnost okamžitého přidání zápisu z jednání
 - (e) možnost zobrazení detailu partnera
3. Správa partnerů
 - (a) možnost editace údajů o firmě, kontaktních osobách a provozovnách
 - (b) možnost vytvoření nové objednávky, včetně možnosti upravit ceny pro konkrétního partnera
 - (c) zobrazení historie objednávek
 - (d) přidání zápisu z jednání
 - (e) zobrazení všech událostí a zápisů z jednání s možností filtrování pomocí komplexního formuláře
4. Typy událostí jsou
 - (a) přihlášení do systému
 - (b) nahrání souboru
 - (c) vytvoření zápisu z jednání
 - (d) vytvoření objednávky
 - (e) poslání hromadného emailu
5. Správa produktů
 - (a) zadání názvu
 - (b) možnost editace popisu produktu pomocí wysiwyg editoru

- (c) přiřazení obrázku
- (d) přiřazení do kategorie
- (e) nastavení maximálního počtu kusů k odběru
- (f) nastavení, zda je produkt v nabídce
- (g) nastavení, zda je produkt nabízen zdarma
- (h) zadání standardní ceny
- (i) zadání rozsahu cen podle počtu kusů
- (j) zadání jednotlivých cen pro jednotlivé partnery
- (k) možnost pozicování produktů v kategorii

6. Správa kategorií

- (a) zadání názvu
- (b) nastavení, zda je kategorie aktivní
- (c) možnost pozicování kategorií

7. Správa uživatelů CRM

- (a) nastavení loginu, jména, příjmení, emailu, hesla, oprávnění a zda je aktivní
- (b) typy oprávnění jsou (obchodní zástupce, marketing a administrátor)

8. Zápisy z jednání

- (a) komplexní formulář pro vyhledání a třídění zápisů z jednání
- (b) možnost přiložení souboru

9. Správa reportů

- (a) komplexní formulář pro vyhledání a třídění různých typů reportů
- (b) typy reportů jsou (Zápisy z jednání, Přehled objednávek, Přehled partnerů vč. kontaktů, Přehled partnerů - pobočky, Události, Přehled kontaktních osob)
- (c) možnost exportu do CSV formátu
- (d) možnost okamžitého tisku

10. Vytvoření hromadného emailu

- (a) zadání textu pomocí wysiwyg editoru
- (b) hrubý filtr příjemců podle krajů
- (c) ruční filtr příjemců podle kontaktních osob
- (d) nastavení oslovení 5. pádem
- (e) rekapitulace
- (f) odeslání

11. Správa doplňujících stránek

- (a) jméno stránky
- (b) URL
- (c) nastavení, zda je aktivní
- (d) editovatelný obsah pomocí wysiwyg editoru

12. Správa nastavení

- (a) emailové adresy pro posílání nových objednávek
- (b) znění informačního emailu o vytvoření objednávky
- (c) znění informačního emailu o potvrzení objednávky
- (d) znění informačního emailu o zrušení objednávky
- (e) obecné oslovení (pro případ, že u kontaktní osoby není vyplněno oslovení 5. pádem)

13. Správa webových stránek

- (a) editování všech stránek pomocí wysiwyg editoru
- (b) možnost vytváření a editace článků do sekcí média a tiskové zprávy

3.5 CRM - obchodník

1. Správa objednávek
2. Správa partnerů
3. Správa reportů
4. Statistiky

3.6 CRM - marketing

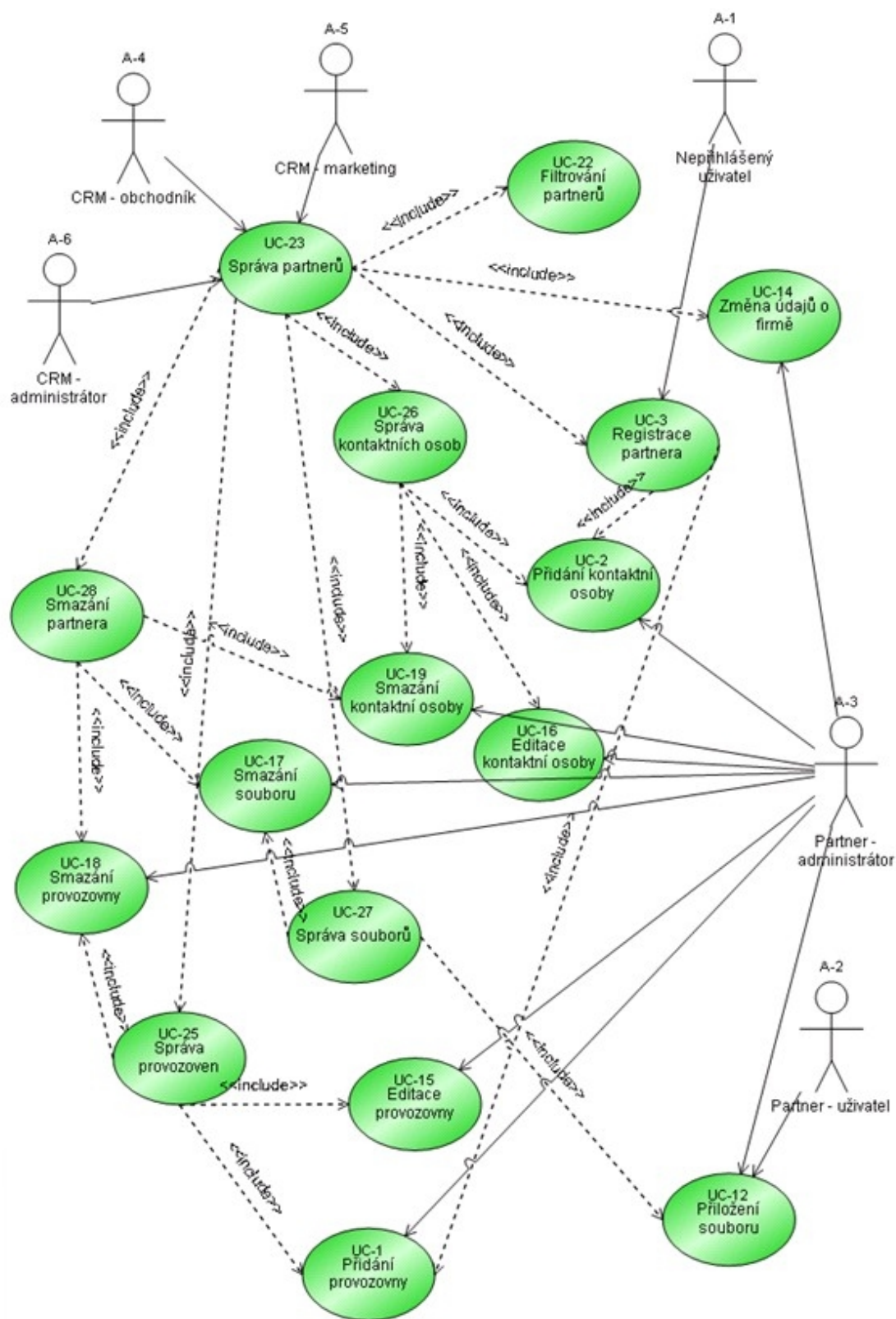
1. Správa objednávek
2. Správa doplňujících stránek
3. Správa partnerů
4. Správa produktů
5. Správa kategorií
6. Správa nastavení
7. Správa reportů

- 8. Zápisy z jednání
- 9. Hromadné emaily
- 10. Statistiky

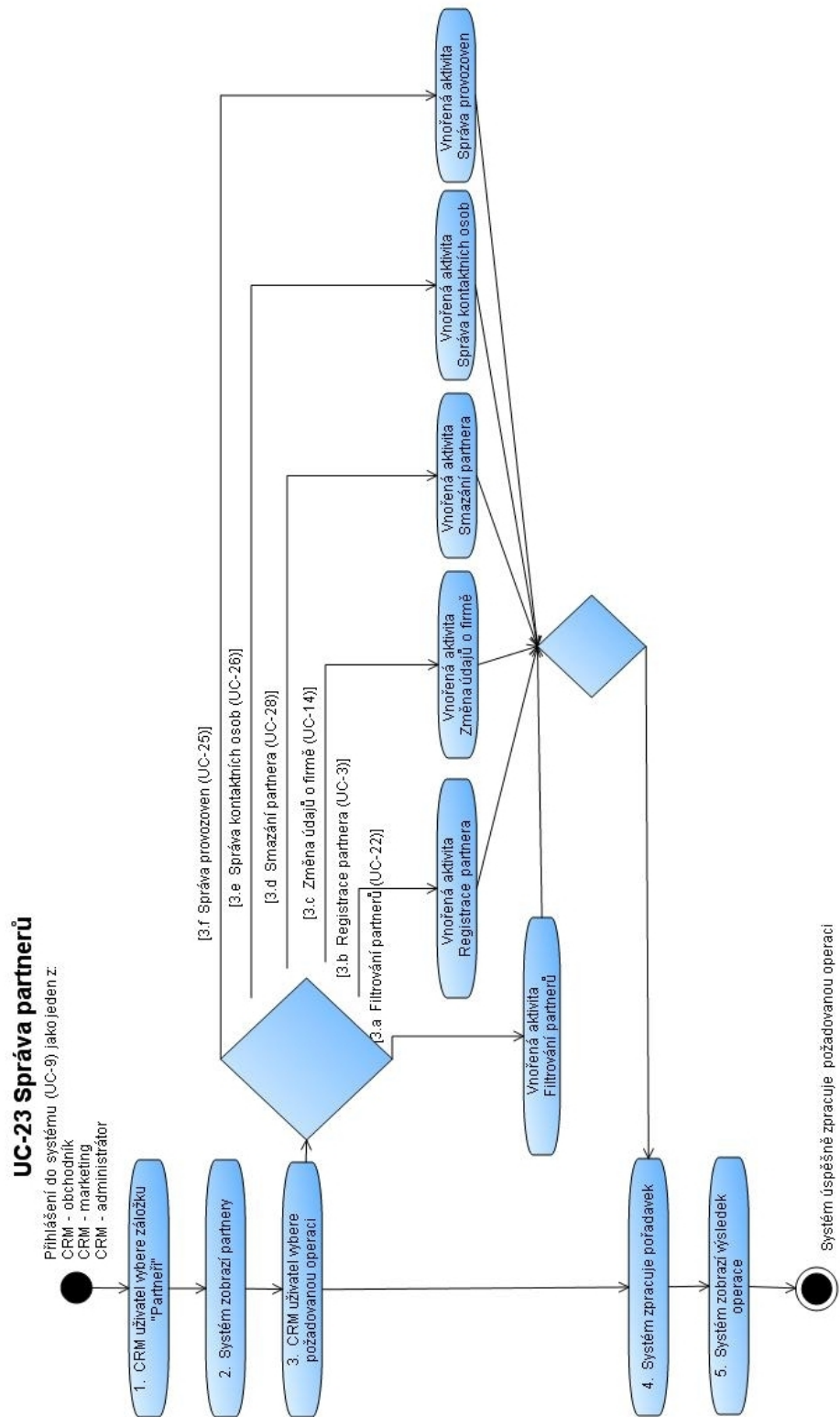
Jako ukázkou use case diagramu jsem si vybral Správu partnerů. V tabulce 4 je popis tohoto případu užití zahrnující všechny informace. Na obrázku 1 můžete vidět vizualizovaný tento rozšířený případ užití pomocí UML, který je zobrazen do nižších úrovní. Na obrázku 2 je zobrazen příslušný diagram aktivit. Finální report s kompletní specifikací, aktéry, případy užití a slovníkem jsem vytvořil v programu CaseComplete v. 2010 a exportoval do wordu. Celý dokument je uložen na DVD, které je součástí této práce.

UC-23 Správa partnerů	
Proces zajišťující vytváření, filtrování, upravování a mazání partnerů.	
Detaily	
Složka: StarCommunications	
Primární aktéři: CRM - administrátor, CRM - marketing, CRM - obchodník	Supporting Actors: Nepřihlášený uživatel, Partner - uživatel, Partner - administrátor
Vstupní podmínky: Přihlášení do systému (UC-9) jako jeden z: CRM - obchodník, CRM - marketing, CRM - administrátor	Záruka úspěchu: Systém úspěšně zpracuje požadovanou operaci
Úroveň: Uživatel	Složitost: Vysoká
Status: Kompletní	Status implementace: Kompletní
Přiřazeno:	Verze:
Tok událostí	
Hlavní scénář: <ol style="list-style-type: none"> 1. CRM uživatel vybere záložku 'Partneři' 2. Systém zobrazí partnery 3. CRM uživatel vybere požadovanou operaci 4. Systém zpracuje požadavek 5. Systém zobrazí výsledek operace 	
Rozšíření: <ol style="list-style-type: none"> 3.a Filtrování partnerů (UC-22) 3.b Registrace partnera (UC-3) 3.c Změna údajů o firmě (UC-14) 3.d Smazání partnera (UC-28) 3.e Správa kontaktních osob (UC-26) 3.f Správa provozoven (UC-25) 	

Tabulka 4: UC – Správa partnerů



Obrázek 1: Diagram užití: Správa partnerů



Obrázek 2: Diagram aktivit: Správa partnerů

4 Analýza a návrh

Na základě případů užití je možné stanovit jednotlivé kroky analýzy a vytvořit analytický model systému. V této fázi se provádí analýza architektury a případu užití, identifikují se budoucí komponenty, řeší se návrh tříd, subsystémů a databáze.

Základním typem je analýza datová a vychází z požadavků na systém. Vyberou se příslušné objekty, určí se jejich atributy a závislosti a navrhne se struktura databáze. Výsledkem je konceptuální model, který obsahuje lineární zápis všech typů entit a jejich atributů, úplný ER diagram, datový slovník a souhrn IO omezení, která se týkají atributů nebo celých entit. Dále se v této fázi provádí funkční a časová analýza.

Vzhledem k rozsáhlosti systému se zde omezím pouze na popis hlavních modulů. Kompletní datový slovník a kompletní třídní diagram se nachází na přiloženém DVD, které je součástí této práce.

Celá aplikace se dá rozdělit do dvou částí. První je webový systém a druhá CRM systém, přičemž obě sdílejí stejnou databázi. S webovým systémem pracuje webový uživatel (nepřihlášený, přihlášený a CRM - administrátor), práci v CRM systému zajišťuje CRM uživatel (administrátor, marketing, obchodník). V následujících sekcích provedu analýzu jednotlivých modulů.

Na obrázcích 3 a 4 můžete vidět kontextový diagram a Data Flow Diagram 0. úrovně.

4.1 Webová část

Ve webové části bude existovat několik hlavních modulů. Nejdůležitější jsou registrace partnerů, on-line nakupování, správa provozoven, správa uživatelů, přikládání dokumentů. V této sekci se zaměřím na popis prvních dvou modulů, které mají nejvyšší prioritu. Ostatní moduly jsou zároveň součástí CRM části.

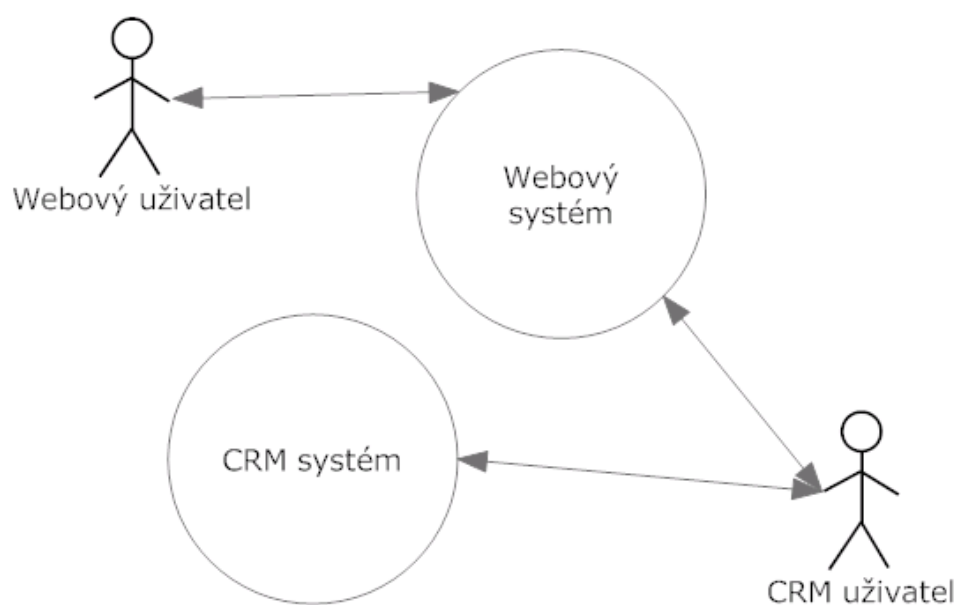
4.1.1 Registrace partnerů

Proces registrace partnerů probíhá v pěti krocích. Prvním krokem po kliknutí na „Registrace“ je vyplnění údajů dané firmy. Uživatel zadá potřebné informace a potvrdí formulář. Systém musí v tuto chvíli provést validaci všech vstupních dat a zkontrolovat, zda již náhodou není firma s uvedeným IČ registrována. Pokud uživatel nevyplní některé povinné údaje, nebo zadá špatný formát informací, systém ho o této skutečnosti upozorní výpisem chybové hlášky u příslušného pole. Až po splnění všech těchto podmínek systém dovolí přejít ke kroku 2, ve kterém je možno zadat informace o provozovnách.

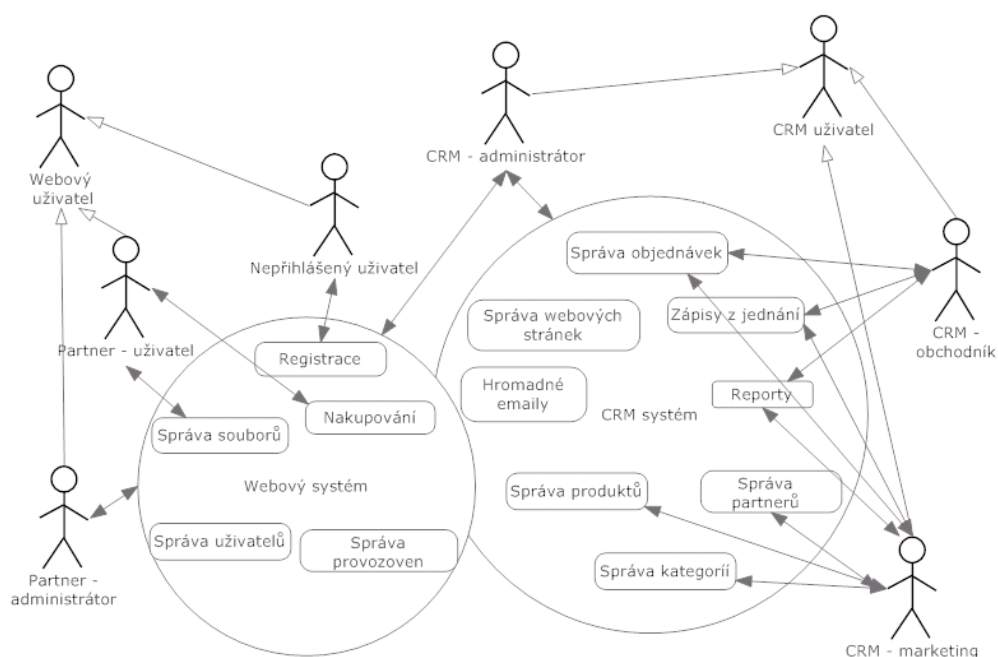
V tomto kroku si může uživatel vytvořit více provozoven, které se automaticky naváží k dané firmě. Systém opět musí zkontrolovat validitu všech údajů a teprve potom může přejít ke kroku 3. Systém musí také zajistit zpětné procházení registrace, tudíž musí ukládat vytvořené údaje do mezipaměti.

V kroku 3 si volí uživatel své jméno, příjmení, telefon, email a další informace ke své osobě. Systém opět zkontroluje validitu emailu a dalších informací.

Ve 4. kroku se volí přístupové údaje. Uživatel si zvolí login, heslo, potvrzení hesla a odešle formulář. Systém musí ověřit, zda ještě v databázi neexistuje uživatel s daným



Obrázek 3: Kontextový diagram



Obrázek 4: DFD – 0. úroveň

loginem a přistoupí k dokončení registrace, kde zobrazí veškeré vyplněné údaje. Pokud by chtěl uživatel libovolné údaje měnit, systém mu to umožní.

Teprve po dokončení registrace může systém uložit údaje z mezipaměti do databáze a poslat potvrzující email na registrovanou osobu o úspěšném přihlášení. Mezipaměť musí být následně vymazána. V tomto modulu budeme potřebovat tyto tabulky:

regions (id, name)

firms (id, firmname, ico, dic, city, street, number, psc, web, region_id, vodafone_card)

users (id, login, email, domains, firstname, lastname, title, position, firm_id, phone)

offices (id, name, phone, city, street, number, psc, firm_id, active)

users_offices (id, user_id, office_id)

Tučně jsou označeny primární klíče, kurzívou cizí klíče a podtrženě atributy, které je nutné indexovat. Nutnost indexace jsem zvolil u těch atributů, podle kterých se bude často vyhledávat nebo třídit. Samozřejmostí je indexace primárních a cizích klíčů.

Vazební tabulka *users_offices* je vytvořena proto, že potřebujeme zajistit aby jeden uživatel mohl spravovat více provozoven a zároveň jedna provozovna mohla sdílet více uživatelů.

Na obrázku 5 můžete vidět diagram tříd pro registraci nové firmy.

4.1.2 On-line nakupování

Modul e-shop je přístupný pouze registrovaným a přihlášeným uživatelům. Systém musí zajistit, aby nikdo jiný nemohl nakupovat. Po přihlášení si uživatel vybere příslušnou kategorii a systém zobrazí všechny aktivní produkty v této kategorii. Uživatel si zvolí počet kusů a může přidat položku do košíku. Systém přepočítá cenu podle počtu kusů.

Při odběru více kusů může mít produkt nastaven jinou cenu. Tento proces se může opakovat a nakonec si uživatel zvolí adresu firmy nebo provozovny, na kterou chce zboží doručit. Systém zaeviduje objednávku a pošle potvrzující email na adresu uživatele a kopii na vedení firmy Star Communications a.s. Entity, které se vyskytují v tomto modulu jsou následující:

categories (id, name, category_id, position, visible)

items (id, name, description, short_description, price, dph, discount, max_count, in_menu, position)

categories_items (id, category_id, item_id)

scopes (id, item_id, from_value, to_value, price)

order_groups (id, name)

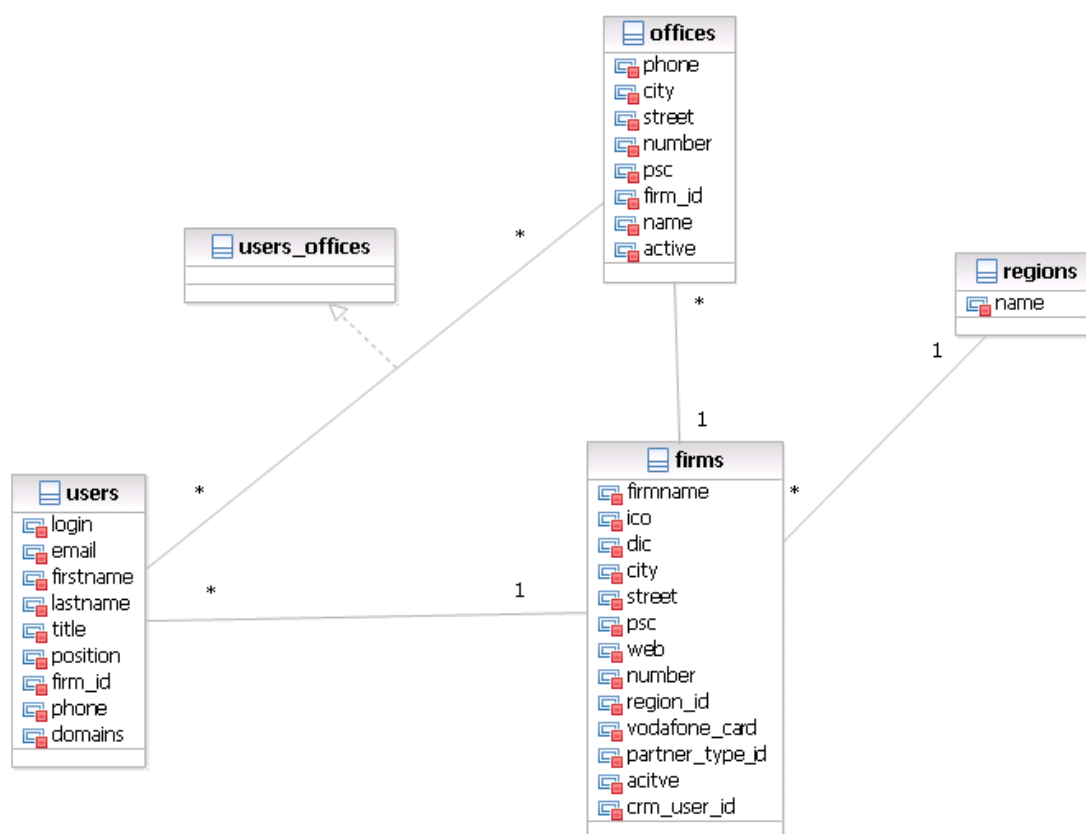
orders (id, sum, state, created_at, payed, user_id, office_id, order_group_id, canceled_at)

shoppings (id, item_id, count, actual_price, order_id)

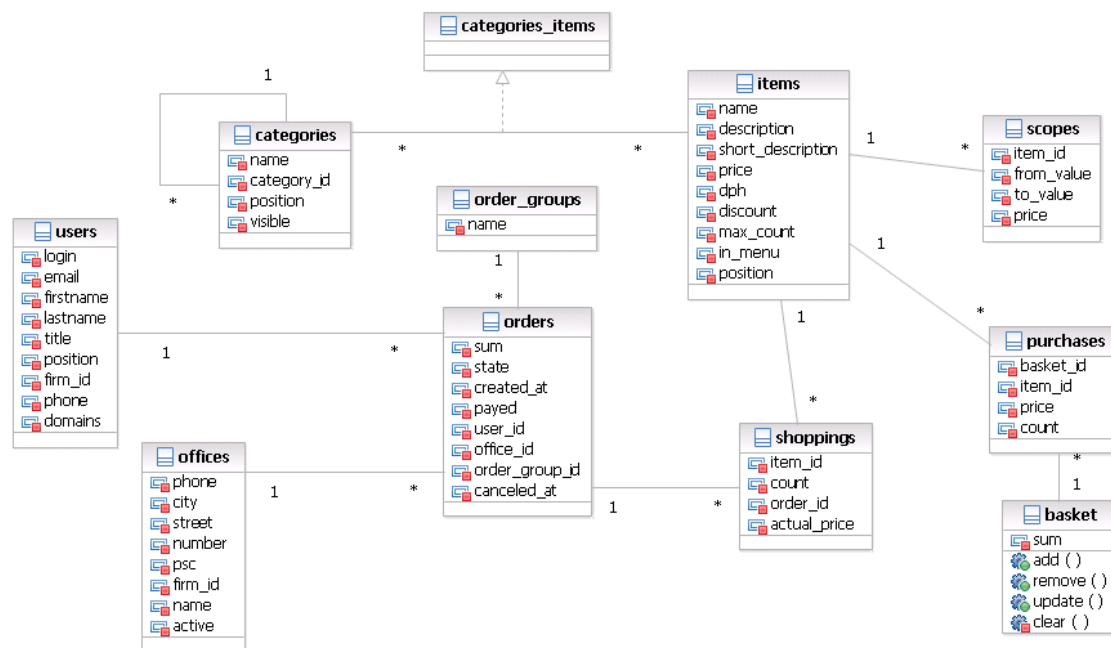
Dále se zde vyskytují již známé tabulky *users* a *offices*. Na obrázku 6 můžete vidět diagram tříd pro on-line nakupování.

Do tohoto modulu patří také nákupní košík, který bude reprezentován vlastní třídou **basket** (sum) a objekty nákupů reprezentované třídou **purchase** (basket_id, item_id, count, price). Platí pro ně vazba 1:N.

Tyto objekty se neukládají do databáze, ale zůstávají v interní paměti v průběhu objednávání. Po dokončení objednávky se odstraní.



Obrázek 5: Třídní diagram: Registrace partnera



Obrázek 6: Třídní diagram: On-line nakupování

4.2 CRM část

V CRM části budou existovat tyto moduly. Správa partnerů, správa objednávek, správa produktů, správa kategorií, hromadné emaily, správa webových stránek, správa doplňujících stránek a reporty. Všechny tyto moduly jsou přístupné pro CRM - administrátora.

4.2.1 Správa partnerů

Oproti webové části, kde může partner upravovat jen své základní údaje, zde bude několik funkcí navíc. Mezi ty nejdůležitější patří přiřazení partnerů do skupin. Budou existovat tři základní skupiny. Dále je možné přiřadit firmě nějakého zástupce ze CRM oddělení, přičemž zástupce může být vždy jen jeden. Další funkcí navíc oproti webové části je nastavení vlastních cen produktů pro jednotlivé partnery.

Vzhledem k těmto vlastnostem se musí rozšířit datová analýza. Tabulky, kterých se týkají změny budou vypadat následovně:

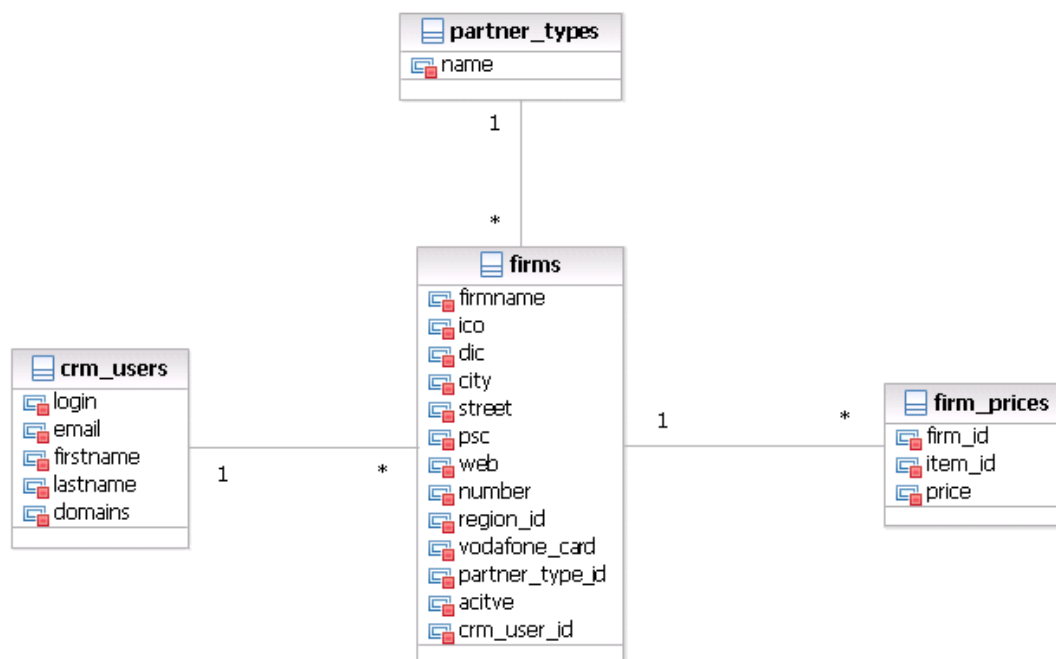
firms (id, firmname, ico, dic, city, street, number, psc, web, region_id, vodafone_card, partner_type_id, active, crm_user_id)

partner_types (id, name)

crm_users (id, login, email, fristname, lastname, domains)

firm_prices (id, firm_id, item_id, price)

Na obrázku 7 je doplněný diagram tříd pro správu partnerů.



Obrázek 7: Doplněný třídní diagram: Správa partnerů

Do atributu *domains* v tabulce *users* a *crm_users* se ukládají práva uživatelů. Vzhledem k tomu, že CRM uživatelé mohou mít více rolí (CRM - administrátor je zároveň i CRM - marketing, atd.), by bylo vhodnější vytvořit novou tabulku pro role a vazební tabulku mezi uživateli a rolemi.

V tomto návrhu však připustím dělitelnost tohoto atributu a budu ukládat do *domains* role uživatelů v pevně daném formátu (tzv. hashmap).

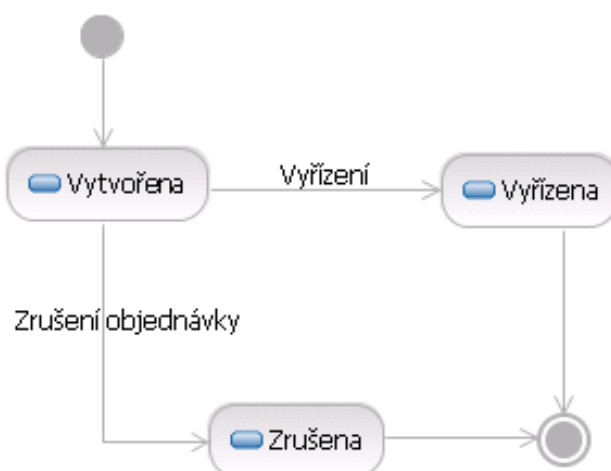
4.2.2 Správa objednávek

Tento modul umožňuje filtrování objednávek podle různých kritérií (za různá časová období, podle krajů, nebo stavu). V CRM části mohou uživatelé, navíc oproti webové části, měnit ceny v průběhu přidávání do košíku. A dále mohou měnit stav objednávky.

Po vytvoření je objednávka ve stavu „Vytvořena“, partner může požádat o zrušení a objednávka se tak přesune do stavu „Zrušena“. Pokud nedojde ke zrušení, objednávka nabude stavu „Vyřízena“. Řešením této stavové analýzy bude přidání atributu „state“ do tabulky „orders“, který bude nabývat příslušných hodnot.

Hodnoty atributu „state“ by měly být typu Integer, protože databáze je lépe zpracovává. Řešením bude namapovat jednotlivé stavy na čísla. „Vytvořena“ = 0, „Zrušena“ = 1 a „Vyřízena“ = 2.

Na obrázku 8 je znázorněn stavový diagram, který znázorňuje životní cyklus entity Objednávka.



Obrázek 8: Stavový diagram: Objednávka

4.2.3 Správa produktů

Tento modul umožňuje vytvářet, editovat a mazat produkty. Tabulky s jejich atributy potřebné k realizaci tohoto modulu jsem již uvedl ve webové části. Zde se zaměřím na vzájemnou interakci mezi těmito objekty při vytváření nového produktu.

Kromě základních informací o produktu si mohou uživatelé CRM vytvářet rozsahy cen pro různé množství koupeného zboží. Dále mohou nastavovat zvláštní ceny pro zvláštní skupinu partnerů. Popis procesu vytváření nového produktu a interakce mezi objekty je znázorněn na obrázku 9.

4.2.4 Hromadné emaily

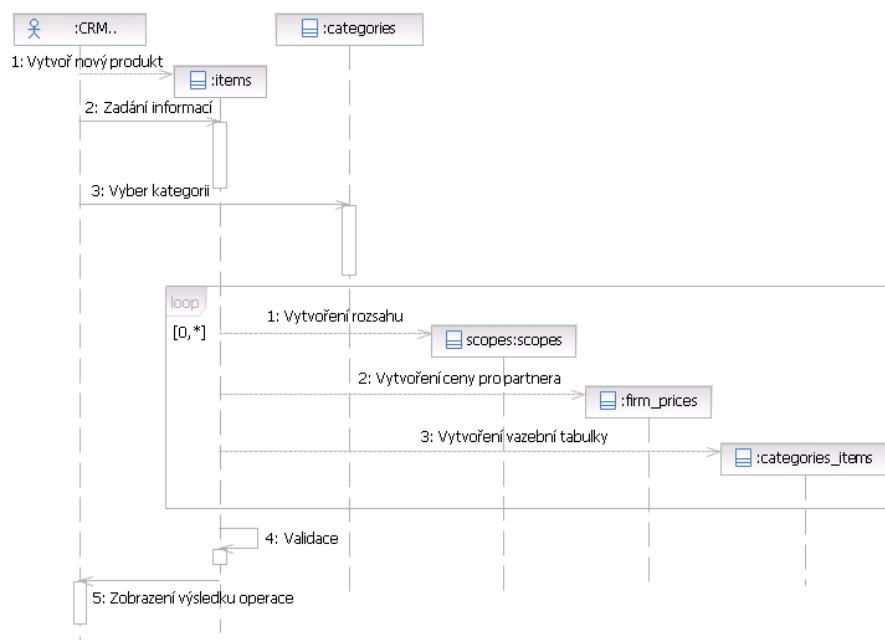
Jedním z požadavků na systém je posílat hromadné emaily. Text emailu se bude vytvářet ve wysiwyg editoru. Měl by tudíž existovat nějaký modul pro tvorbu textu, který bude předávat data do modulu hromadných emailů. Propojení těchto modulů bude jak na databázové, tak aplikační vrstvě.

Základem bude tato tabulka:

mailings (id, subject, from_name, from_email, name, vocative, text)

Do atributu text se bude ukládat výstup z wysiwyg modulu ve formě volného textu. Vzhledem k možnosti přikládání obrázků do textu bude muset existovat tabulka *images* pro centrální úschovnu obrázků. Tyto obrázky se nahrají na server a do textu emailu se přilinkují jako tag s příslušnými parametry.

Atribut *vocative* bude uchovávat defaultní oslovení jako je např. „Vážený uživateli“. Pro zpětnou vazbu, aby uživatelé věděli, komu přišel jaký email, je potřeba vytvořit vazební tabulku mezi *mailings* a *users*.



Obrázek 9: Sekvenční diagram: Vytvoření produktu

users_mailings (*id*, *mailing_id*, *user_id*)

Na obrázku 10 můžete vidět třídní diagram, který znázorňuje propojení modulů pro wysiwyg editor a posílání emailů.

4.2.5 Zápisy z jednání

Administrátor a marketing může v sekci CRM vytvářet a spravovat zápisy z jednání k jednotlivým firmám. CRM - obchodník bude mít přístup pouze pro filtrování zápisů. U každého zápisu bude potřeba evidovat následující entity a atributy.

dealings (*id*, *firm_id*, *plan_date*, *real_date*, *sc_id*, *office_id*, *dealing_type_id*, *note*)

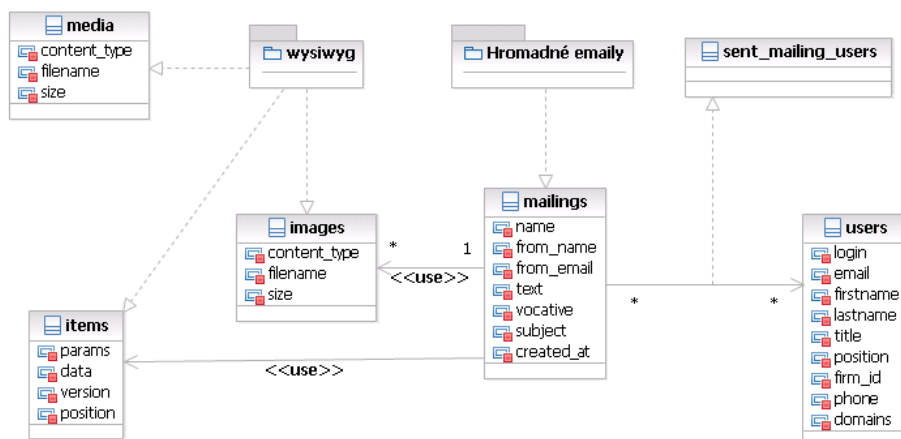
Datum plánování *plan_date* bude možno edivat pouze několik hodin po vytvoření zápisu. Do atributu *sc_id* bude uloženo id uživatele, který zápis vytvořil, nesmí mít tudíž nulovou hodnotu.

Při vytváření zápisu může uživatel dynamicky vytvářet kontaktní osoby, osoby přítomné ze Star Communications, typy zápisů a přiložené soubory. Soubory se budou přikládat na hardisk do složky *dealing_files*. Pro zjištění, zda je soubor obrázků bude sloužit atribut *is_image*. Obrázky se budou totiž zobrazovat přímo na webu a ostatní soubory povedou na download. Entity s vazbou k zápisům z jednání jsou tyto:

dealing_crm_users (*id*, *dealing_id*, *crm_user_id*) V této tabulce budou všechny osoby ze Star Communications přítomné u zápisu z jednání.

dealing_files (*id*, *name*, *is_image*, *extension*) Tato tabulka bude uchovávat informace o přiložených souborech.

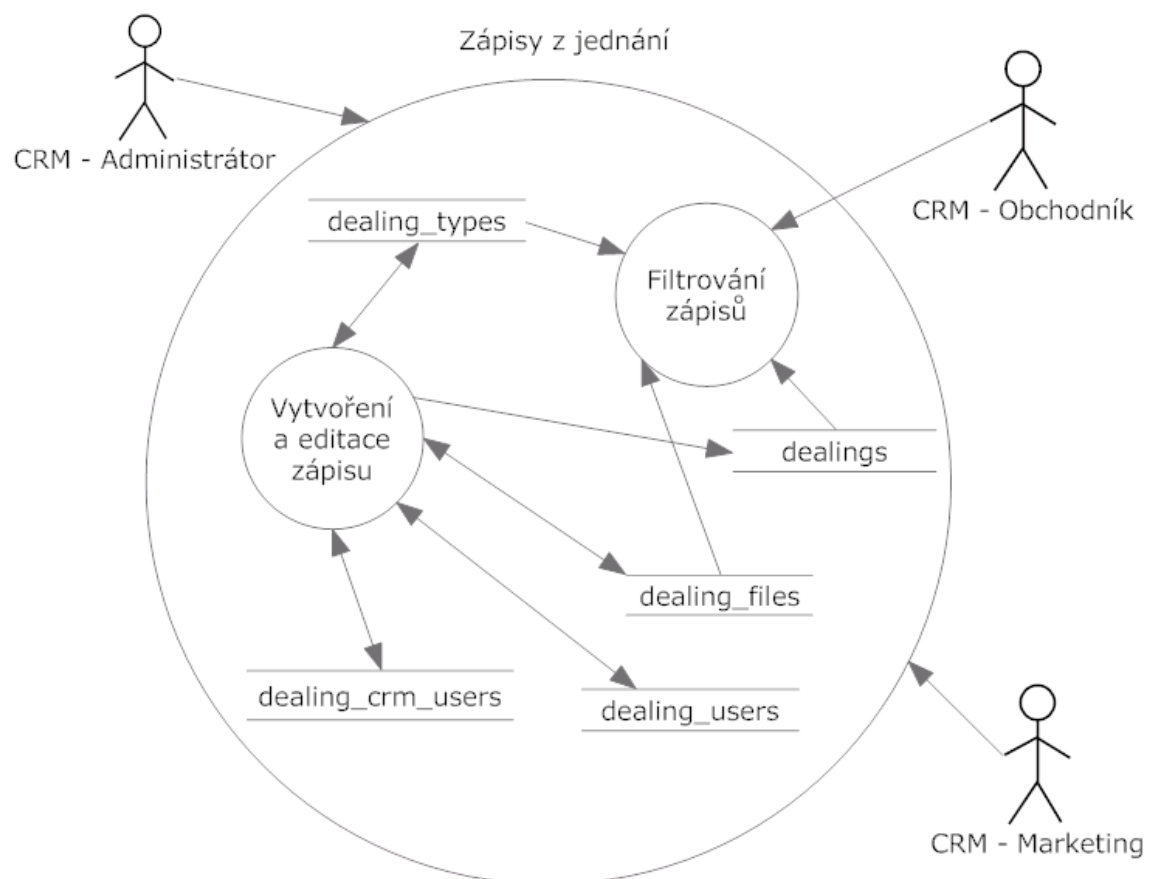
dealing_types (*id*, *name*) Zvláštní tabulka pro typy zápisů.



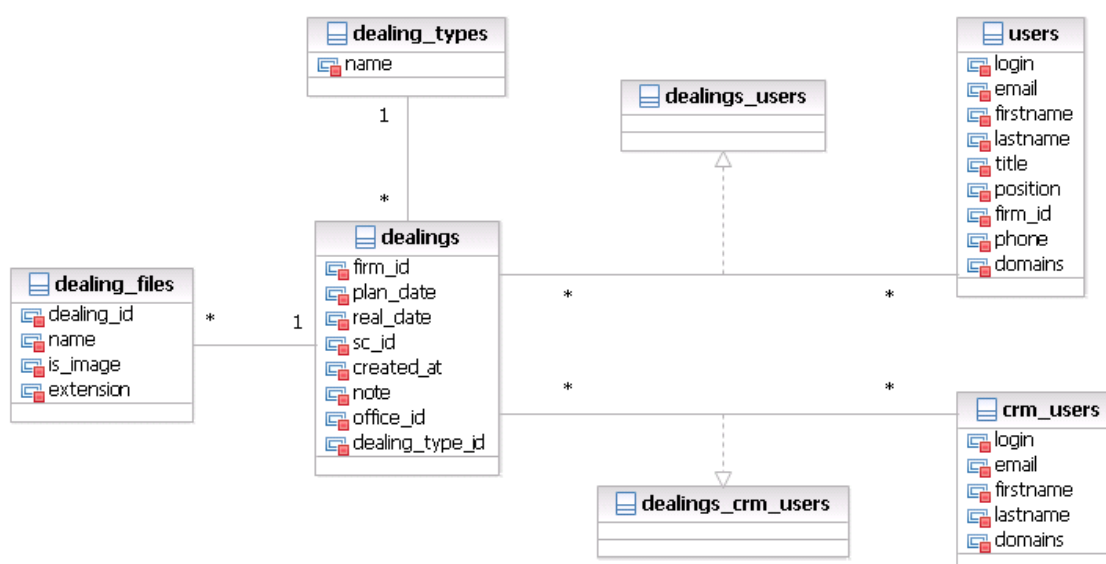
Obrázek 10: Třídní diagram: Hromadné emaily

dealing_users (id, dealing_id, user_id) Tabulka pro zaznamenání všech účastníků dané firmy na daném zápise.

Diagram datových toků pro Zápisy z jednání je zobrazen na obrázku 11. Vazby mezi tabulkami jsou zachyceny v třídním diagramu na obrázku 12.



Obrázek 11: DFD_1: Zápisy z jednání



Obrázek 12: Třídní diagram: Zápisy z jednání

5 Implementace

Implementace projektu má být provedena v jazyce Ruby on Rails. Knížek o Ruby je dnes již hodně, zmíním zde proto jen hlavní filozofii tohoto jazyka. Zaměřím se na důležité části, které jsou pro implementaci tohoto portálu zásadní. Informace z následující kapitoly jsem čerpal z [2, 3].

5.1 Ruby

První verze tohoto jazyka byla uveřejněna v roce 1995. Tvůrcem Ruby je jediný člověk – Yukihiro „Matz“ Matsumoto. Ruby je dynamický interpretovaný scriptovací plně objektově orientovaný programovací jazyk. Má velmi jednoduchou syntaxi (inspirovaný Perlem), je poměrně lehký na naučení a dostatečně výkonný.

Ruby podporuje dynamické typování, kdy typy nejsou přiřazeny proměnným, ale až jejich hodnotám. Dále podporuje automatickou správu paměti založenou na algoritmu mark and sweep. Díky široké podpoře regulárních výrazů se hodí pro zpracování textů, nebo parsování dokumentů. Hodí se pro tvorbu CGI scriptů, což jsou programy schopné komunikovat s WWW serverem. V Ruby je možné psát i GUI aplikace.

Programovací jazyk Ruby má rozsáhlou standardní knihovnu, je přenositelný mezi všemi hlavními platformami, má centrální úložiště tzv. balíčků přes RubyGems, podporuje výjimky, umožňuje přetížení operátorů a mnoho dalších funkcí, které usnadňují práci. Existuje řada alternativních implementací, jako jsou YARV, JRuby, MacRuby atd.

Krásné zdůvodnění, proč si zrovna vybrat jazyk Ruby jako programovací jazyk, má na svém blogu [4] Karel Minařík, který podle mě patří k nejvýznamnějším širitelům tohoto jazyka u nás i v zahraničí, a z něhož jsem čerpal nejvíce informací.

„Věřím, že – alespoň do jisté míry – je smyslem života být šťastný. Na základě tohoto přesvědčení je Ruby navrženo tak, že je nejenom snadné, ale i zábavné v něm programovat. Ruby vám umožňuje soustředit se na kreativní stránku programování, a nepřiděluje vám další starosti.“

Yukihiro Matsumoto, předmluva k prvnímu vydání knihy Programming Ruby.

Ruby skutečně přináší do světa programování nové myšlenky a principy, se kterými je radost pracovat. Vyjadřovací prostředky Ruby se podobají spíše přirozenému jazyku. Ruby má svoji syntaxi, gramatiku i slovní zásobu.

Matsumoto se odvolává přímo na hypotézu jazykového relativismu, která ve své radikální podobě v teorii Ludwiga Wittgensteina tvrdí, že to, co nemohu v jazyce vyjádřit, nemohu ani myslet (*Filosofická zkoumání*, §243 a následující). Vykládá ji tak, že čím pohodlněji se mohu vyjádřit, tím pohodlněji mohu i myslet.

Objektovost jazyka Ruby je vyjádřena ve výpise 1. Toto zřetězení umožňuje psát čitelný a úsporný kód. Ukázka kódu v jazyce Ruby je ve výpise 2. Tento zápis se skutečně podobá mluvenému slovu, dokonce i se správným slovosledem.

Navíc v Ruby není potřeba ukončovat příkazy středníkem, nebo psát závorky při volání metod apod. Tím se skutečně odlišuje od jiných jazyků a stává se tak přehlednější.

1 + 2 je vlastně zavolání metody + objektu 1 s argumentem 2:
1.+(2)

Výpis 1: Ruby – vše je objekt

`print self.email unless self.email.blank?`

Výpis 2: Program v jazyce Ruby

Cituji z blogu [4]:

„Úspornost a čitelnost kódu, snadno zapamatovatelná syntaxe bez obtížné ‚interpunkce‘ — to vše slouží k tomu, aby se programátor mohl soustředit na ‚myšlenku‘ a ne na to, jak ji vyjádřit. Lze říci, že všechny ostatní, ryze technické vlastnosti Ruby jsou podřízeny tomuto principu.“

Tyto základní principy ovlivnily návrh celého frameworku Rails, který je v Ruby napsán.

5.2 Rails

V této kapitole budou popsány hlavní myšlenky frameworku Rails. Hlavním zdrojem byl blog [5].

„Toto je sněhová vločka. Vaše aplikace není jedna z nich. Většina věcí, které většina lidí dělá, není nijak unikátní. Vaše potřeby nejsou nijak ‚zvláštní‘.“
David Heinemeier Hansson, přednáška na konferenci Future Of Web Applications, únor 2006

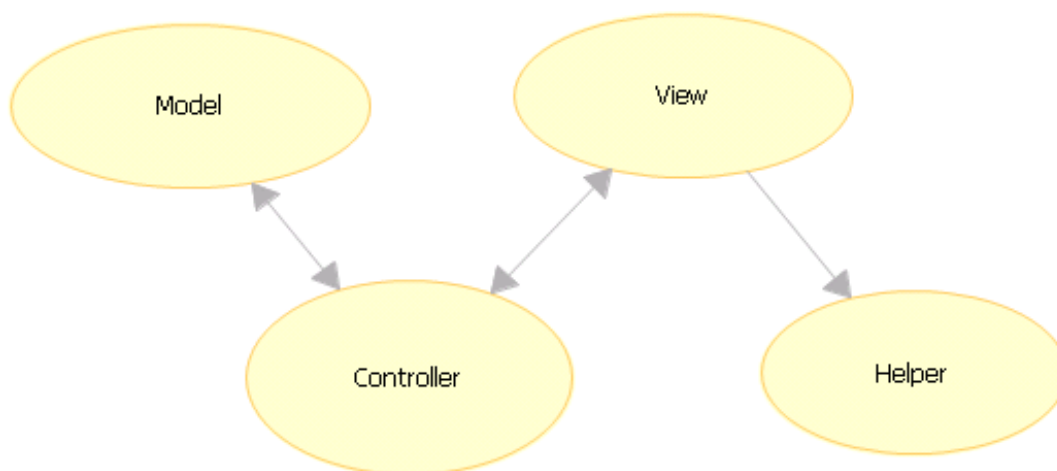
S touto klíčovou myšlenkou vytvořil právě David Heinemeier Hansson framework Rails jako „vedlejší produkt“ při práci na projektu Basecamp. Všechny webové aplikace jsou si totiž velice podobné: pracují s daty nad databází, vypisují je do HTML a zpracovávají je zpět pomocí formulářů. Základní pravidlo návrhu Rails znamená např. to, že programátor nastavuje v aplikaci pouze ty prvky, které se liší od běžného nastavení.

„Less code!“ („Méně kódu!“) a „Don’t repeat yourself!“ („Nepiš to dvakrát!“) jsou hlavní filosofie Rails. Vše by mělo být deklarováno na jednom místě a využito pro ušetření psaní kódu navíc.

Rails jsou postaveny na principu MVC, tedy Model-View-Controller. Cituji z blogu [5]:

„MVC je ustálený návrhový vzor v softwarovém vývoji, způsob, jak transparentně oddělit různé části aplikace (lhostejno či webové nebo desktopové) podle toho, jakou úlohu v ní hrají.

Model se stará o čtení a zapisování dat a o manipulaci s nimi (Vrat’ mi seznam osob, kde název zaměstnavatele je ‚Škoda Auto‘, seřazený vzestupně podle příjmení); data předává controlleru.



Obrázek 13: MVC a Helper – předávání dat

Controller (česky někdy zvaný řadič) se stará o příjem vstupů ze strany uživatele aplikace (Proved' akci definovanou pro kliknutí na odkaz 'Telefonní seznam') a distribuci výstupů (Načti stránku s identifikátorem 'telefonni-seznam').

View (česky zvaný též pohled, v kontextu webu též šablona) obsahuje grafické rozhraní aplikace samotné, s definovanými ovládacími prvky (tlačítka, roletky, odkazy, ...) a 'dírami', do nichž se leje obsah, který do view předává controller."

Někdy se také k této trojici připojují tzv. Helpery, které usnadňují práci s Views. Píšeme do nich pomocné metody pro zpracovávání textových výstupů a jiné metody, které odlehčují logiku z Views. Na obrázku 13 je znázorněna komunikace mezi jednotlivými součástmi.

Mezi klíčové součásti Rails patří:

1. **Model:** Stará se o mapování dat z relační databáze na objekty (object-relational mapping), které je zajištěno díky Active Record
2. **Controller:** Mapuje URL na řídicí prvky uvnitř aplikace (metody) a předává data do views (HTML šablon)
3. **View:** Stará se o zobrazovací logiku získaných dat a zpětnou interakci s controllers pomocí formulářů
4. **Helper:** Obsahuje pomocné metody pro práci s daty, formuláři, Ajaxem, formátování textů atd.

5. **Konzole:** Umožňuje přímou interakci s modely na příkazové řádce (script/console)
6. **Sdílení kódu:** Centrální úložiště v podobě gemů (hotové aplikace nebo knihovny Ruby, které je možno instalovat, odinstalovat a spouštět) a pluginů
7. **Skripty pro generování kódu:** např. scaffold, migration a další (script/generate)

5.2.1 Active Record

Active Record je návrhový vzor, který převádí tabulky v databázi na třídy, jednotlivé řádky na instance těchto tříd (objekty) a sloupce na příslušné atributy. Díky implementaci anglického slovníku, nebo explicitnímu rozšíření, namapuje třídu (Category) na tabulku (categories). Navíc umožňuje definovat vztahy mezi tabulkami, které jsou známé z konceptuálního schématu (Category has many :items). Podle obrázku 6 na straně 25 ukážu, jak by vypadala implementace vztahu mezi kategoriemi a položkami viz výpis 3.

```
class Category < ActiveRecord::Base
  belongs_to :parent, :class_name => "Category", :foreign_key => :category_id

  has_many :subcategories,
    :class_name => "Category",
    :foreign_key => :category_id,
    :dependent => :destroy

  has_many :categories_items
  has_many :items, :through => :categories_items

  validates_presence_of :name, :message=>"_can_not_be_empty."
end

class CategoriesItem < ActiveRecord::Base
  belongs_to :category
  belongs_to :item
end

class Item < ActiveRecord::Base
  has_many :categories_items
  has_many :categories, :through => :categories_items
end
```

Výpis 3: Rails – Active Record

Pokud libovolná třída v programu dědí z ActiveRecord::Base, automaticky ji Rails mapují na databázovou tabulku.

Vazba *belongs_to* říká Active Record, aby hledal v této tabulce cizí klíč, který je definovaný atributem *:foreign_key*. Pokud tento atribut nebyl definovaný, hledají Rails automaticky cizí klíč podle názvu vazby (v našem případě *parent_id*) a namapují nalezený objekt na třídu definovanou v atributu *:class_name*, tedy Category. Opět, pokud by nebyl atribut definovaný, přiřadily by Rails výsledný objekt ke třídě Parent.

Vazba *has_many* je obdobou vazby *belongs_to*, ale obráceně (1:N). Říká, aby ActiveRecord hledal ve třídě, která je definována vazbou (nebo atributem *:class_name*) všechny

ID	name	category_id
1	Hlavní kategorie	NULL
2	Podkategorie hlavní	1
3	Podkategorie druhé úrovně	2

Tabulka 5: Categories

ID	in_menu	position
1	1	1
2	1	3
3	0	1
4	1	2

Tabulka 6: Items

údaje, které mají cizí klíč (*:foreign_key*) roven aktuálnímu *:id* objektu. Výsledné údaje jsou vráceny jako pole ActiveRecord objektů.

Vazba *has_many :through* je známá jako vazba M:N v relačním schématu, kde je potřeba vazební tabulka (klíčové slovo *through*). Alternativou je použití *has_and_belongs_to_many*, která mapuje tuto vazbu bez nutnosti deklarovat vazební třídu. V následujícím příkladě jsou uvedeny příklady použití ActiveRecord nad výše uvedenými třídami.

Jednou z nutností při tvorbě správné webové aplikace je validace vstupních dat. V ActiveRecord jsou zabudované funkce pro automatickou tvorbu validací. Pokud uvedeme např. *validates_presence_of :name, :message => " can not be empty."* jako ve výše uvedeném příkladě a budeme chtít vytvořit objekt jako v příkladu 4, pak první metoda *save* se pokusí uložit objekt do databáze, ale vrátí *false*, protože objekt se nepodaří zvalidovat. Druhá metoda *save!* je agresivnější. Ta přímo vyvolá výjimku s příslušnou validační zprávou.

```
c = Category.new(:name=> "")
c.save #=> false
c.save! #=> ActiveRecord::RecordInvalid: Validation failed : Name can not be empty.
```

Výpis 4: Příklad použití validace

Předpokládejme strukturu databáze, viz obrázek 6 na straně 25 a třídy definované ve výpise 3 na straně 35. Mějme v databázi mimo jiné tabulky s údaji, viz tabulky 5, 6 a 7.

ID	item_id	category_id
1	1	1
2	1	2
3	2	2
4	3	1
5	4	1

Tabulka 7: Categories_items

Potom následující operace vrátí příslušné výsledky (uvedené za #=>), viz výpis 5.

```
c = Category.find(1)
c.attributes
#=> {"id"=> 1, "name"=>"Hlavní_kategorie", "category_id" => nil}

c.subcategories.length
#=> 1

c.items.length
#=> 3

c.items.find (: all ,: conditions=>{:in_menu=>true}).length
#=> 2

i = Item.find(2)
i.categories.first.parent.name
#=> "Hlavní_kategorie"

new_item = Item.create(:in_menu => true, :position => 0)
#=> #<Item id: 5, in_menu: true, position: 0>

c.items << new_item
#=> [#<CategoriesItem id: 6, item_id: 5, category_id: 1>]

items = Item.find_all_by_in_menu(true, :select=>"id,position", :order=>"position_DESC")
#=> [#<Item id: 2, position: 3>, #<Item id: 4, position: 2>, #<Item id: 1, position: 1>, #<Item
id: 5, position: 0>]
```

Výpis 5: Příklad ActiveRecord

Z výše uvedeného příkladu je patrné, že Active Record osvobodí programátora od psaní „zbytečných“ SQL dotazů, které jsou ve své podstatě stejné a umožňuje se soustředit na hlavní cíle.

Poznámka 5.1 V některých případech, například kvůli silné optimalizaci nebo v případě velmi složitých dotazů, je nutné si dotaz napsat sám, proto Active Record nabízí metodu *find_by_sql*, které můžeme předat vlastní SQL dotaz.

5.2.2 Action Controller a Routování

Základem pro zpracování webových požadavků v Rails jsou tzv. Action Controllers. Skládají se z jedné nebo více akcí, které po vykonání mohou zavolat další akci nebo zobrazit HTML stránku. Akce jsou definovány jako veřejné metody v controllers, které jsou automaticky přístupné na webovém serveru pomocí routování.

Na příkladě ve výpisu 6 je znázorněn jednoduchý controller, který má definované tři akce. První akce *index* provede načtení všech produktů z databáze a zobrazí pohled (view), který bude hledat ve složce *app/views/products/index.html.erb*.

Druhá akce *create* provede vytvoření produktu s atributy uloženými v parametru *product* a zavolá v zápětí akci *index*, která teprve zobrazí výše zmíněný view.

Třetí akce *list* načte všechny aktivní produkty a poté zobrazí view *index*. Neprovede ale opětovné volání metody jako akce *create*.

```
class ProductsController < ActionController::Base
  def index
    @products = Product.all
  end

  def create
    Product.create(params[:product])
    redirect_to :action => "index"
  end

  def list
    @products = Product.find(:all, :conditions=>{:active => true})
    render :action => "index"
  end
end
```

Výpis 6: Příklad na ActionController

Vizitkou výsledné aplikace je i její prezentace z vnějšku. Dosti opomíjený faktor, na který hodně programátorů zapomíná je URL. Skutečnost je taková, že vznikají dosti nečitelné a dlouhé odkazy, které aplikaci jen škodí. Uživatel aplikace by měl podle URL snadno odhadnout, jaký druh obsahu může najít na dané adrese. Tento princip je založen na tzv. směrování (anglicky routing), které zajišťuje, aby aplikace věděla, kde mají směřovat konkrétní požadavky.

V Rails existují pravidla pro routování, která se definují v souboru *config/routes.rb*. Konvence, která určuje, který controller zpracuje požadavek, kterou metodu použije a s jakými parametry bude operovat, vychází ze základního principu, viz výpis 7.

```
map.connect ':controller /: action /: id'
```

Výpis 7: Ukázka routování

Takže pokud dostane aplikace požadavek např. */articles/show/1*, pak Rails budou vědět, že mají vytvořit instanci controlleru *articles* a zavolat jeho metodu *show* s parametrem *id=1*.

Princip routování v Rails není nikterak složitý. Zmíním zde ještě, že existují tzv. *pojmenované routes*. Jejich výhoda tkví v přehlednosti a hlavně v tom, že je lze použít ve views a využít pro ušetření psaní dalšího kódu.

Více informací ohledně Action Controller a routování je možné nalézt zde [7, 8].

5.2.3 ActiveSupport a ActionView

Rozšíření jazyka Ruby o pomocné metody, které přináší řadu zjednodušení, existují v jádře ActiveSupport. Toto rozšíření doplňuje třídy „Array“, „Date“, „Hash“, „Numeric“, „String“ a je přidáno přímo do jazyka Ruby a je tudíž dostupné pro všechny instance těchto tříd.

Ve výpise 8 jsou uvedeny některé metody pro práci s výše uvedenými objekty.

```
[ 'Brambory', 'Mléko', 'Jablka' ].to_sentence(:connector => 'a') #=> "Brambory, Mléko a Jablka"
[1, 2, 3, 4, 5, 6, 7].in_groups_of(3) {|group| puts group}
#=> ["1", "2", "3"]
    ["4", "5", "6"]
    ["7", nil, nil]

Time.now.beginning_of_month #=> Mon Mar 01 00:00:00 +0100 2010

{"name" => "Rob", "years" => "28"}.to_options #=> {:years=>"28", :name=>"Rob"}

2.even? #=> true
3.days.ago #=> Tue Mar 23 13:42:07 +0100 2010

"active_record".camelize #=> "ActiveRecord"
"Žlutoučkový kůň ůpěl".parameterize('.') #=> "Zlutoucky_kun_upel"
"sheep".pluralize #=> "sheep"
```

Výpis 8: Ukázka ActiveSupport

Další skupinou rozšíření tříd Rails je ActionView. Tyto metody jsou ovšem dostupné pouze ve views a slouží k zjednodušení formátování textů a generování HTML kódů.

Poznámka 5.2 Pomocí příkazu *include* lze zpřístupnit metody ActionView i pro controllers a modely.

Ve výpise 9 jsou uvedeny některé metody z ActionView.

```
link_to ("Google", "http://google.com") #=> <a href="http://google.com">Google</a>

number_to_currency(12345, :unit => 'Kč_', :separator => ',', :delimiter => '_')
#=> "Kč_12_345,00"

number_to_human_size 125952 #=> "123_KB"

distance_of_time_in_words_to_now(Time.now.yesterday) #=> "yesterday"
```

Výpis 9: Ukázka ActionView

Další pomocné metody pro práci ve views si můžeme vytvořit sami díky jednoduchému mechanismu helperů. Najdeme je ve složce *app/helpers* a platí pro ně jednoduché pravidlo: *application_helper.rb* obsahuje metody dostupné pro všechny views a ostatní helpery, např. *products_helper.rb* bude obsahovat metody pouze do views pro controller *app/controllers/products_controller.rb*.

5.2.4 Wysiwyg

Pro práci s wysiwyg editorem nabízí Rails plugin, který je snadno instalovatelný a jednoduchý na použití, viz [9]. Tento plugin je postaven na editoru TinyMCE (Javascript WYSIWYG Editor) a je snadno integrovatelný do každé Rails aplikace.

Pro nainstalování javascriptů a css je potřeba spustit příkaz *rake simple_cms:install*. Pro správnou funkčnost je potřeba mít nainstalované další čtyři pluginy, které lze snadno doinstalovat spuštěním *rake simple_cms:install_dependencies*:

- *attachment_fu*
- *responds_to_parent*
- *acts_as_versioned*
- *coderay*

Nakonec se vygenerují potřebné tabulky (*ruby script/generate simple_cms_migrations*), spustí se migrace (*rake db:migrate*) a plugin je připraven k použití.

Pro načítání a úpravu obrázků (pro komunikaci s ImageMagick) je nutné mít nainstalovaný jeden z těchto tří obrázkových procesorů:

- ImageScience - umožňuje pouze změnu velikosti obrázku
- RMagick - umožňuje více komplexnějších operací s obrázky a správu paměti
- minimagick - je jednodušší na paměť než RMagick, protože spouští příkazy ImageMagick přímo v shellu

Jak je patrné, Ruby on Rails jsou velice silný nástroj pro tvorbu webových aplikací. Díky jednoduché syntaxi jazyka Ruby, konvenčním pravidel, návrhového vzoru Active Record, který je možno rozšířit o pluginy jako jsou např. *acts_as_tree*, *acts_as_taggable*, apod., směrovacího mechanismu HTTP požadavků a dalších rozšíření nám umožňují vytvářet kvalitní aplikace v krátkém čase.

6 Nasazení a testování

Výsledný projekt byl nasazen na testovací server `http://starcomm.rstage.ataxo.cz/`. Parametry tohoto serveru jsou následující:

- 2x Intel(R) Xeon(TM) CPU 2.80GHz (2799.23-MHz 686-class CPU), 4GB RAM
- OS: FreeBSD + passenger, apache, mysql

Na pozadí aplikace běží proces, který se stará o průběžné mazání sessions z databáze. Aplikace je uložena na SVN a po úpravách stačí spustit pár příkazů (`svn up`, `rake db:migrate` a restart serveru) a změny jsou ihned viditelné v prohlížeči.

Plán testů je určen pro projekt starcommunications.cz. Primárním cílem plánu je otestovat základní funkcionalitu programu. Dále pak výkonnostní testování internetového obchodu, který slouží k distribuci produktů firmy Star Communications a.s. a zátěžové testování CRM části.

Testování proběhlo na výše uvedeném testovacím serveru ve třech fázích: manuální, funkcionální a výkonnostní.

6.1 Manuální testování

V manuálním testování byly testovány tyto vlastnosti: vzhled, přihlašování a nakupování.

6.1.1 Vzhled

Cílem tohoto testovacího případu je ověření vzhledu stránek.

Předpoklady:

1. Libovolný webový prohlížeč.
2. Připojení k internetu.
3. Tester nemusí být přihlášený a nemusí znát přístupové údaje.

Akce: Na stránce `http://starcomm.rstage.ataxo.cz/` zkontroluje pozici a velikost obrázků.

Očekávaný výsledek: Funkční zobrazení požadované stránky. Ikony budou ve správných velikostech a na správném místě. Dílčí očekávané výsledky jsou zahrnuty ve skriptu testovacího případu.

6.1.2 Přihlašování

Cílem tohoto testovacího případu je ověření přihlašování do systému a to jak s chybnými tak se správnými přístupovými údaji.

Předpoklady:

1. Libovolný webový prohlížeč.

2. Připojení k internetu.
3. Tester nemusí být přihlášený a nemusí znát přístupové údaje.

Akce: Tento test simuluje přihlášení se správnými a chybnými přístupovými údaji. Na stránce <http://starcomm.rstage.ataxo.cz/> zadá do přihlašovacího boxu nejprve chybné údaje, zkontroluje výpis chybové hlášky, poté se přihlásí se správnými údaji a zkontroluje informační hlášku o úspěšném přihlášení.

Očekávaný výsledek: Funkční zobrazení požadované stránky. Vypsání chybové hlášky v případě neúspěšného přihlášení. Vypsání informační hlášky v případě správného přihlášení.

6.1.3 Nakupování

Cílem tohoto testovacího případu je ověření možnosti nakupování ve dvou případech: Jako přihlášený a nepřihlášený uživatel.

Předpoklady:

1. Libovolný webový prohlížeč.
2. Připojení k internetu.
3. Tester nemusí být přihlášený a nemusí znát přístupové údaje.

Akce:

Na stránku <http://starcomm.rstage.ataxo.cz/produkty/divoka-karta> se připojí jako nepřihlášený uživatel a otestuje zobrazení příslušného textu. V druhém případě se připojí na stejnou stránku jako přihlášený uživatel a zkontroluje zobrazení formuláře pro nakupování.

Očekávaný výsledek: Funkční zobrazení požadované stránky. Vypsání požadovaného textu v případě nakupování bez přihlášení. Zobrazení formuláře pro nakupování v případě přihlášeného uživatele.

6.2 Funkcionální testování

V rámci tohoto testování proběhly dva základní testy.

6.2.1 Registrace partnerů

Cílem tohoto testovacího případu je ověřit funkčnost registrace partnerů přes webový formulář.

Předpoklady:

1. Libovolný webový prohlížeč.

2. Připojení k internetu.

Akce: Registrace má 5 kroků. Pokud předáme v parametru číslo, test provede pouze požadovaný počet kroků. Test provede inicializaci hodnot z datapoolu pro každý krok, přičemž otestuje validační hlášky u každého vstupního pole.

Skript zajišťuje následující funkčnost:

1. Spuštění aplikace `http://starcomm.rstage.ataxo.cz/crm`
2. Spuštění ručního vkládání
3. Cyklus pro opakovaný test hodnot(Firma). Počet kroků je dán počtem záznamů v datapoolu
 - (a) Vložení hodnot z datapoolu. Zde mohou nastat dvě varianty:
 - i. Jedná se o negativní test, kontroluje se, zda byly zobrazeny správně elementy, které obsahují chybu
 - ii. Jedná se o pozitivní test, pokračuje se na další krok
 - (b) Pokud se jednalo o pozitivní test, je spuštěn další script
4. Cyklus pro opakovaný test hodnot(Provozovny). Počet kroků je dán počtem záznamů v datapoolu
5. Cyklus pro opakovaný test hodnot(Osoba). Počet kroků je dán počtem záznamů v datapoolu
6. Cyklus pro opakovaný test hodnot(Uživatel). Počet kroků je dán počtem záznamů v datapoolu
7. Aplikace je ukončena

Očekávaný výsledek: Vstupní data a očekávané výsledky jsou uloženy v souboru „SC_FTC001_Registrace.rftxdp“ tzv. datapoolu v projektu „Starcomm“ ve workspace „RFT_Starcommunications“ na přiloženém DVD. Očekává se vypsání validačních upozornění, zobrazení povinných údajů a celkově úspěšný průběh registrace.

6.2.2 CRM - partneři

Cílem tohoto testovacího případu je ověřit manuální hledání v sekci CRM - partneři

Předpoklady:

1. Libovolný webový prohlížeč
2. Připojení k internetu

Akce: Při vyhledávání se ve formuláři vyberou hodnoty(kraj, obchodní zástupce, město, třídít podle, atd.). Test provede zjištění, zda se všechny atributy ve formuláři zachovají i po stránkování a zda po kliknutí na tlačítko „default“ se formulář opět vynuluje.

Skript zajišťuje následující funkčnost:

1. Spuštění aplikace `http://starcomm.rstage.ataxo.cz/crm`
2. Sekce partneři
3. Spuštění ručního vyhledávání
4. Cyklus pro opakované stránkování. Počet kroků je defaultně 2
 - (a) Kontrola, zda je formulář dobře vyplněn
 - (b) Kontrola aktivity stránek
5. Kontrola vymazání formulářových údajů
6. Aplikace je ukončena

Očekávaný výsledek: Vstupní data a očekávané výsledky jsou uloženy v souboru „SC_FTC002_Crm.rftxdp“ takzvaném datapoolu v projektu „Starcomm“ ve workspace „RFT_Starcommunications“. Očekává se zachování filtračních kritérií při stránkování, aktivita stránek a následné vymazání formulářových polí.

6.3 Výkonnostní testování

V rámci tohoto testování proběhly tři základní testy.

6.3.1 Procházení

Skript simuluje zákazníka internetového obchodu, který pouze prochází nabízené zboží a nic si neobjednává. V cyklu prohlíží kategorie a detaily produktů.

Očekávaný výsledek: Kód HTTP odpovědi všech dotazovaných položek musí být „200 OK“.

6.3.2 Objednávání

Skript simuluje zákazníka internetového obchodu, který si objedná jednu položku z nabízených produktů. Simuluje přidání „x“ položek daného produktu do košíku. Hodnota „x“ se mění v iteracích dle hodnot v datapoolu. Produkt se také mění v iteracích.

Očekávaný výsledek: Kód HTTP odpovědi všech dotazovaných položek musí být „200 OK“.

6.3.3 CRM

Skript simuluje uživatele v sekci CRM, který prochází a edituje partnery.

Očekávaný výsledek: Kód HTTP odpovědi všech dotazovaných položek musí být „200 OK“.

6.3.4 Zátěžový test

Zátěž je generována výše uvedenými třemi skripty „SC_PTC001_load_test.testsuit“, „SC_PTC002_shopping.testsuit“ a „SC_PTC003_crm.testsuit“ v poměru 60:30:10.

Aktuálně je zátěž testována na 5 uživatelích (licenční omezení). V případě dostupnosti dalších licencí je třeba zátěž navýšit alespoň na 50 uživatelů.

Očekávaný výsledek: Test bude hodnocen jako úspěšný, pokud výsledná průměrná doba odpovědi na stránku nebude větší než 1,5 vteřiny a maximální doba jakékoliv stránky nepřekročí 5 vteřin.

6.4 Závěrečná zpráva

6.4.1 Manuální testování: 100%

V manuálních testech bylo provedeno celkem 8 kontrolních bodů. Všechny prošly úspěšně.

6.4.2 Funkcionální testování: 82%

Ve funkcionálních testech bylo provedeno celkem 38 kontrolních bodů. Z toho 31 prošlo úspěšně a 7 neúspěšně. Jedná se především o méně závažné chyby týkající se barvy rámečku kolem chybových polí, které se v Internet Exploreru vyšší verze zobrazují s jiným stylem, než byly testovány.

Ve shodě s plánem testů byl proveden pouze omezený rozsah testování. Nicméně v těchto testech aplikace fungovala dle očekávání a proto lze předpokládat, že množství zbývajících chyb je minimální.

6.4.3 Výkonnostní testování: 66%

V rámci ověřování výkonu byl proveden zátěžový test internetového obchodu. Podmínky testu nebyly zcela splněny. Průměrná doba odpovědi je sice 1,2 s, ale maximální doba čekání na stránku byla 8,3 s. Z analýzy výsledků testů vyplývá, že systém je nadměrně zatěžován při objednávání položek. Je třeba optimalizovat dotazy nad databází a zvážit možnosti tzv. cachování.

Vysoká doba odezvy při načítání stránky může být dána také tím, že testy byly prováděny na testovacím serveru, který je intenzivně více zatěžován, než produkční server <http://www.starcommunications.cz>.

6.4.4 Celkové zhodnocení

Projekt nemá žádné přetrvávající vady a kvalita vyvinutého produktu je velice uspokojivá. Projekt může být v současné podobě distribuován zákazníkovi.

7 Závěr

Cílem této práce bylo navrhnout a implemetovat internetový portál pro firmu Star Communications a.s. Během práce na tomto projektu jsem si prohloubil teoretické i praktické znalosti z oblasti vývoje softwarového díla pomocí RUP a pronikl jsem více do nitra Ruby on Rails. Také jsem nastudoval mnoho pluginů, které usnadňují práci v tomto frameworku.

Výsledná aplikace bude hojně využívána zaměstnanci firmy Star Communications a.s. zejména v části CRM, která jim umožňuje sledovat statistiky, reporty a dynamicky upravovat hlavní části systému. Webová část umožňuje zejména on-line nakupování a správu informací týkajících se dané firmy. Celý systém je zdokumentován. Dokumentace k systému je dostupná na přiloženém DVD.

Pomocí testů jsem ověřil funkčnost a výkonnost výsledné aplikace, provedl jsem optimalizaci SQL dotazů a v této podobě byl portál distribuován zákazníkovi. Nyní se aplikace těší z využití na stránkách <http://www.starcommunications.cz>, což bylo cílem.

8 Reference

- [1] IBM Corporation, *Writing Good Use Cases*, Cupertino, January 2007.
- [2] Ruby (programming language),
[http://en.wikipedia.org/wiki/Ruby_\(programming_language\)](http://en.wikipedia.org/wiki/Ruby_(programming_language))
- [3] Ruby (programovací jazyk),
[http://cs.wikipedia.org/wiki/Ruby_\(programovací_jazyk\)](http://cs.wikipedia.org/wiki/Ruby_(programovací_jazyk))
- [4] Ruby on Rails a revoluce ve vývoji pro web. Část první: Ruby,
<http://blog.karmi.cz/2007/6/16/co-je-ruby-on-rails-cast-1>
- [5] Ruby on Rails a revoluce ve vývoji pro web. Část druhá: Active Record,
<http://blog.karmi.cz/2007/6/16/co-je-ruby-on-rails-cast-2>
- [6] Ruby on Rails a revoluce ve vývoji pro web. Část třetí: Controller a view,
<http://blog.karmi.cz/2007/7/3/co-je-ruby-on-rails-cast-3.html>
- [7] Rails Routing from the Outside In,
<http://guides.rubyonrails.org/routing.html>
- [8] Class ActionController::Base,
<http://api.rubyonrails.org/classes/ActionController/Base.html>
- [9] pullmonkey / simple_cms,
http://github.com/pullmonkey/simple_cms
- [10] Railscasts - Free Ruby on Rails Screencasts,
<http://railscasts.com/>
- [11] Ruby on Rails guides,
<http://guides.rubyonrails.org/>

A Obsah DVD

Adresář	Popis
/doc/api	Dokumentace
/doc/diagrams	Kompletní projekt s diagramy pro program IBM Rational Software Architect
/doc/files	Dokument vize, datový slovník, kompletní report z programu Case Complete, kompletní třídní diagram
/doc/testing	Kompletní projekt a report z testovacích programů od firmy IBM
/doc/text	Text diplomové práce
/install	Instalační soubory
/src	Zdrojové soubory aplikace

Tabulka 8: Obsah DVD

B Dokument vize

B.1 Předmluva

Firma Star Communications a.s. se rozrůstá a v současné době nabízí desítky produktů pro různé typy partnerů. Prozatím má informační systém, který umožňuje pouze statické procházení stránek. S přibývajícím počtem partnerů vzrůstá i nutnost jejich spravování a organizování. Dále má potřebu umožnit distribuci produktů na internetu a on-line nakupování.

Výše zmíněné potřeby vedly k nutnosti vytvořit nový informační systém, který bude natolik efektivní, že umožní on-line registraci partnerů, spravování uživatelů a provozoven, nakupování zboží přes e-shop a komplexní správu pomocí CRM systému.

B.1.1 Cíl

Cílem tohoto dokumentu je definice a analýza hlavních vlastností a potřeb systému pro firmu Star Communications. Ukazuje potřeby tzv. Stakeholderů, cíle uživatelů a proč tyto potřeby existují.

B.1.2 Rozsah

Funkčnost systému je rozdělena do následujících úloh:

Webový systém:

1. On-line registrace partnerů
2. Přihlašování partnerů a správa provozoven
3. Evidence a správa uživatelů ve webové části
4. Rozdělení uživatelů do dvou skupin(admin a uživatel)
5. Možnost nakupování přes e-shop
6. Možnost nahrávat soubory

CRM systém:

1. Evidence všech zaměstnanců firmy Star Communications a jejich rozdělení do skupin s různými právy
2. Správa partnerů a provozoven
3. Evidence a správa kategorií a produktů
4. Správa objednávek
5. Editace webové části přes wysiwyg editor

6. Vytváření doplňujících stránek
7. Vytváření reportů podle různých kritérií
8. Posílání hromadných emailů
9. Nastavení textů pro automatické emaily

B.2 Nastavení

B.2.1 Obchodní příležitost

Tento systém je vytvořen pouze pro firmu Star Communications a její potřeby. Žádná jiná firma, kromě přihlášených partnerů ve webové části, nebude se systémem pracovat.

B.2.2 Prohlášení o problému

Problém: Žádná správa partnerů, nemožnost on-line distribuce produktů, neexistence automatických prostředků pro tvorbu reportů a statistik. **Ovlivňuje:** Každého zaměstnance firmy Star Communications a přihlášeného partnera. **Dopad:**

1. Nemožnost spravovat partnery
2. Snížení kvality provozu
3. Žádná možnost správy webových stránek
4. Nemožnost spravovat produkty
5. Nemožnost spravovat objednávky
6. Žádný přístup ke statistickým údajům

Úspěšné řešení: Informační systém, který bude umožňovat registrovaným partnerům pohodlnou správu provozoven, uživatelů a on-line nakupování. CRM systém, nad kterým budou mít plnou kontrolu pracovníci firmy Star Communications a který bude poskytovat statistické údaje a možnost spravovat partnery, objednávky, produkty a webové stránky pohodlně a efektivně.

B.2.3 Prohlášení o produktu

Pro: Star Communications a.s.

Kdo: Každý registrovaný partner a zaměstnanci firmy Star Communications

Co: Softwarový produkt, založený na technologii Ruby on Rails

Umožní: Efektivní správu partnerů, produktů, objednávek, webových stránek a statistik

Rozdíl: Neefektivní manuální řešení správy partnerů a webu

Náš produkt:

1. Umožňuje partnerům registraci, správu uživatelů, provozoven a on-line nakupování
2. Nabízí rychlé a efektivní řešení správy partnerů, produktů...
3. Poskytuje příjemné webové rozhraní

B.3 Stakeholders a uživatelský popis**B.3.1 Demografie**

Vedení firmy Star Communications souhlasilo s implementací projektu, který pokryje potřeby a požadavky nového systému. Finální produkt poskytuje dostatečné služby přiměřené kvality na pokrytí všech potřeb pro správu partnerů, CRM systému a webového systému.

B.3.2 Stakeholder Přehled

Viz tabulka 9

B.3.3 Přehled uživatelů

Viz tabulka 10

B.3.4 Uživatelské prostředí

Existují tři skupiny uživatelů v sekci CRM a dvě skupiny uživatelů ve webové části. Sekce CRM využívá databázi z webové části pro určení statistik, různých výpisů, třídění a zpracovávání objednávek. Z tohoto pohledu je nutné zimplementovat možnost stránkování výpisů pro rychlejší odezvu a možnost exportu do CSV formátu nebo tisku v uživatelsky příjemné podobě.

Webovou část využívají hlavně partneři pro vytváření objednávek a správu provozoven, souborů a uživatelů. Obě části jsou implementovány v jazyce Ruby on Rails 2.2.2, instalovány na Linuxovém serveru s databázovým systémem MySQL. Tato platforma bude používána pro další vývoj systému, který pokryje požadavky firmy Star Communications.

B.3.5 Stakeholder Profily**Skupina CRM - administrátorů**

Reprezentuje: Skupina pověřených osob firmy Star Communications.

Popis: Nejvýše postavení uživatelé firmy Star Communications.

Typ: Expert.

Jméno	Reprezentuje	Role
CRM - administrátor	Skupina lidí, kteří jsou určeni firmou Star Communications ke správě celého CRM systému.	Zodpovědný za aktuálnost informací v systému (ceny produktů, webové stránky, tiskové zprávy).
CRM - marketing	Zaměstnanci firmy Star Communications se všemi právy kromě editace webových stránek.	Spravuje partnery, objednávky, produkty, hromadné emaily, zápisy z jednání...
CRM - obchodník	Zaměstnanci Star Communications s omezenými právy.	Má přístup pouze k partnerům, objednávkám, reportům a statistikám.
Partner - administrátor	Registrovaná kontaktní osoba nebo vedoucí partnerské firmy.	Má přístup k celému webovému rozhraní. Může vytvářet další kontaktní osoby, provozny...
Partner - uživatel	Kontaktní osoba s omezenými právy. Má login a heslo.	Může se přihlašovat a vytvářet objednávky.
Nepřihlášený uživatel	Běžný člověk, který není přihlášený.	Má roli pozorovatele celého vnějšího systému. Může se registrovat.
Manažer	Vedení firmy	Řídí firmu jako celek. Určuje ceny, přijímá nové lidi.

Tabulka 9: Přehled stakeholderů

Jméno	Popis	Stakeholder
CRM - administrátor	Spravuje celý CRM systém.	CRM - administrátor.
CRM - marketing	Spravuje partnery, objednávky, produkty, atd.	CRM - marketing.
CRM - obchodník	Spravuje partnery, objednávky, reporty a statistiky.	CRM - obchodník.
Partner - administrátor	Má přístup k celé webové části.	Partner - administrátor.
Partner - uživatel	Má přístup pouze k objednávkám a souborům.	Partner - uživatel.
Nepřihlášený uživatel	Libovolný návštěvník portálu.	Nepřihlášený uživatel.
E-mail server	Stará se o posílání emailů	

Tabulka 10: Přehled uživatelů

Zodpovědnost: Zodpovědnost je dána tím, že vytváří nové uživatele a přiřazuje jim oprávnění. Musí dbát na to, aby nebyla práva zneužita.

Úspěšná kritéria: Jeho znalosti systému musí být intuitivní.

Zapojení: Může být potenciální tester systému.

CRM - marketing

Reprezentuje: Skupina zaměstnanců Star Communications, s patřičnými právy.

Popis: Mají přístup do celého systému kromě správy CRM uživatelů.

Typ: Expert.

Zodpovědnost: Jejich zodpovědnost je dána skutečností, že spravují všechny objednávky systému a musí dbát na to, aby byly včas vyřízeny a správně z kalkulovány. Dále mají možnost vytvářet nové partnery a spravovat stávající včetně jejich uživatelů a provozu.

Úspěšná kritéria: Jeho znalosti systému musí být intuitivní.

Zapojení: Může být potenciální tester systému.

CRM - obchodník

Reprezentuje: Skupina zaměstnanců Star Communications, s patřičnými právy.

Popis: Má přístup pouze v sekci CRM k partnerům a objednávkám.

Typ: Pokročilý uživatel.

Zodpovědnost: Zodpovědnost je dána tím, že vytváří objednávky a spravuje partnery a jejich provozu.

Úspěšná kritéria: Jeho znalosti systému musí být intuitivní.

Zapojení: Může být potenciální tester systému.

Partner - administrátor

Reprezentuje: Skupina uživatelů, kteří zaregistrují novou firmu.

Popis: Většinou se jedná o vedoucí firmy.

Typ: Středně pokročilý uživatel.

Zodpovědnost: Jejich zodpovědnost je dána tím, že mohou vytvářet nové uživatele, provozu a objednávky ve webové části.

Úspěšná kritéria: Jeho znalosti systému musí být intuitivní.

Zapojení: Může být potenciální tester systému.

Partner - uživatel

Reprezentuje: Skupina uživatelů, kteří patří k firmě.

Popis: Většinou se jedná o zaměstnance firmy.

Typ: Středně pokročilý uživatel.

Zodpovědnost: Jejich zodpovědnost je dána tím, že mohou vytvářet nové objednávky ve webové části.

Úspěšná kritéria: Jeho znalosti systému musí být intuitivní.

Zapojení: Může být potenciální tester systému.

Výsledky: Skupina uživatelů, kteří zaregistrují novou firmu.

B.3.6 Uživatelské profily

Skupina uživatelů je shodná se skupinou stakeholderů. Všechny vlastnosti jsou shodné.

B.3.7 Potřeby uživatelů

Viz tabulka 11

B.4 Přehled o produktu

B.4.1 Perspektiva produktu

Finální produkt umožňuje všechny výše zmíněné potřeby stakeholderů a uživatelů. Odděluje webovou a CRM sekci. Zaznamenává a odděluje přístupová práva uživatelů a zabráňuje přístupu do nepovolených sekcí. Obě sekce spojuje společná databáze.

B.4.2 Souhrn schopností

Viz tabulka 12

B.4.3 Předpoklady a závislosti

Závislosti vyplývají z použití wysiwyg editoru: Musí být nainstalovaný procesor pro úpravu obrázků(ImageMagick) a knihovna do Rails(RMagick), která tento procesor umí propojit s frameworkem Rails. Žádné další závislosti nejsou.

B.4.4 Náklady a ceny

Žádné další příplatky nejsou.

B.4.5 License a instalace

Instalace nevyžaduje žádnou další konfiguraci, protože projekt poběží na serveru, kde je již vše připraveno. Pro přístup webovým klientům musí být koncový počítač vybaven kompatibilním webovým prohlížečem.

B.5 Funkce produktu

B.5.1 On-line registrace partnerů

Toto je hlavní funkce systému. Znamená, že z libovolného koncového počítače, který je připojen na internet lze registrovat nového partnera.

B.5.2 On-line nakupování

Tato vlastnost znamená, že libovolný partner může nakupovat zboží prostřednictvím internetu.

Potřeba	Priorita	Týká se	Dosavadní řešení	Navrhované řešení
Registrace nových partnerů	Vysoká	Nových partnerů	Stará manuální správa	Automatizované a optimalizované registrace s validací formulářů.
Nastavení přístupových práv	Vysoká	Registrovaných partnerů	Žádné dosavadní řešení	Vytvoření systému s automatickými právy, který umožní oddělit jednotlivé webové části.
On-line nakupování	Vysoká	Registrovaných partnerů	Žádné dosavadní řešení	Vytvoření eshopu
Přikládání souborů	Nízká	Registrovaných partnerů	Žádné dosavadní řešení.	Vytvoření rozhraní pro upload souborů
Vytváření CRM uživatelů	Vysoká	Admin sekce v CRM	Žádné dosavadní řešení	Zabránění přístupu nepovolaným osobám do určených sekcí
Distribuce produktů na webu - eshop a správa produktů	Vysoká	Marketing sekce	Dosavadní řešení umožňuje pouze statickou distribuci produktů s jednotnými cenami	Vytvoření eshopu a administrační části pro dynamické vytváření kategorií, produktů a cen
Správa objednávek	Vysoká	CRM sekce	Manuální řešení	Vytvoření administrační sekce pro automatickou správu objednávek
Statistiky, reporty	Vysoká	Marketing sekce	Manuální tvorba statistik	Automatické generování statistik podle různých kritérií
Správa partnerů, provozoven a uživatelů	Vysoká	CRM sekce	Manuální řešení	Systém pro automatizované úpravy
Třídění partnerů do skupin	Vysoká	Marketing sekce	Žádné dosavadní řešení	Automatické nastavování skupiny pro partnery a třídění partner

Tabulka 11: Uživatelské potřeby

Výhoda zákazníka	Podporující funkce
On-line registrace nových partnerů	Webová část obsahuje formulář pro registraci nového partnera. Ověřuje validaci jednotlivých polí a umožňuje nastavit hned v počátku provozovny a uživatele
On-line nakupování	Webová část umožňuje registrovaným a přihlášeným partnerům on-line nakupování produktů. Dynamicky přepočítává cenu a umožňuje nastavit dodací adresu podle vybrané provozovny
Zavedení přístupových práv odděluje jednotlivé funkce v CRM části a systém je tak přehlednější	Díky podpoře uživatelských práv systém kontroluje uživatele, jestli má přístup do jednotlivých sekcí
Statistické údaje jsou dostupné od libovolného data	Systém eviduje jednotlivé události partnerů(vytvoření objednávky, partnera, provozovny, nahrání souboru, přihlášení...) a nabízí tyto události filtrovat v CRM sekci
Jednoduchá editace eshopu	CRM sekce umožňuje jednoduchou správu kategorií, produktů a partnerů. Dokáže nastavit ceny zvlášť jednotlivým partnerům, nebo podle skupin. Editace produktů je ve formě wysiwyg editoru, což usnadňuje vytváření popisu produktů
Editace stránek	Modul pro správu stránek umožňuje vytváření a editování stránek pomocí wysiwyg editoru
Posílání hromadných emailů	V CRM sekci mohou uživatelé filtrovat partnery a posílat jim hromadné emaily
Správa objednávek	CRM sekce umožňuje jednoduché filtrování objednávek, okamžité vyřízení nebo zrušení a poslání příslušného emailu

Tabulka 12: Souhrn schopností

B.5.3 Správa produktů a kategorií

Tato vlastnost dovoluje pověřeným osobám v sekci CRM přidávat a upravovat produkty eshopu. Umožňuje nastavit různé ceny pro různé typy partnerů a popis produktu pomocí wysiwyg editoru.

B.5.4 Uživatelské skupiny

Tato vlastnost znamená, že každý uživatel má určité oprávnění. Systém hlídá veškeré přístupy a kontroluje tato oprávnění a buď umožní nebo neumožní přístup do požadované sekce.

B.5.5 Statistické údaje a reporty

Tato vlastnost umožňuje uživatelům Star Communications vidět přehledy a statistiky jednotlivých partnerů za různá období (Co kdy nakoupili, kdy se kdo přihlásil apod.). Veškeré údaje lze exportovat do formátu CSV.

B.5.6 Správa partnerů

Tato vlastnost dovoluje v sekci CRM přidávat a upravovat partnery pro případ, že by se chtěl někdo zaregistrovat telefonicky, nebo zapomněl heslo.

B.5.7 Správa objednávek

Tato vlastnost umožňuje uživatelům Star Communications evidovat jednotlivé objednávky a řadit je do vyřízených, zrušených apod.

B.5.8 Posílání hromadných emailů

V CRM sekci mohou uživatelé filtrovat partnery a posílat jim hromadné emaily napsané ve wysiwyg editoru.

B.5.9 Správa webových a doplňujících stránek

Modul pro správu stránek umožňuje vytváření a editování stránek pomocí wysiwyg editoru.

B.6 Omezení

Vzhled musí být založen na dosavadním vzhledu stránek firmy Star Communications.

B.7 Rozsah kvality

Výkon - Musí být vynaloženo maximum výkonu, protože projekt bude k dispozici 24 hodin denně. Tolerance chyb - netoleruje se žádná chyba. Použitelnost - systém musí být intuitivní a jednoduchý na používání pro všechny zainteresované uživatele.

B.8 Priority

Nejvyšší prioritu má: registrace partnerů, správa uživatelů, správa objednávek a produktů, nastavení přístupových práv, správa reportů, posílání hromadných emailů a distribuce produktů na webu. S nižší prioritou přichází možnost přikládání souborů

B.9 Dokumentace

B.9.1 Uživatelský manuál

Rozsáhlý manuál by měl být vytvořen, aby podporoval všechny vlastnosti systému. Popis by měl pokrýt všechny funkce systému. Také by měly být zdokumentovány funkce systému na pozadí pro administrátory.

B.9.2 Design

Design musí být založen na stávajícím typickém designu firmy Star Communications.